



Chrystiano Barbosa de Souza Araújo

**Geração de malha de elementos finitos com
inserção de restrições a posteriori**

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática da PUC-Rio

Orientador: Prof. Waldemar Celes Filho

Rio de Janeiro
Setembro de 2012



Chrystiano Barbosa de Souza Araújo

**Geração de malha de elementos finitos com
inserção de restrições a posteriori**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela comissão examinadora abaixo assinada.

Prof. Waldemar Celes Filho

Orientador

Departamento de Informática — PUC-Rio

Prof. Marcelo Gattass

Departamento de Informática – PUC-Rio

Prof. Luiz Fernando Campos Ramos Martha

Departamento de Engenharia Civil – PUC-Rio

Prof. Hélio Côrtes Vieira Lopes

Departamento de Informática – PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro

Técnico Científico — PUC-Rio

Rio de Janeiro, 21 de Setembro de 2012

Todos os direitos reservados. Proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

Chrystiano Barbosa de Souza Araújo

Graduou-se Bacharel em Ciências da Computação pela Universidade Candido Mendes - Campos. Foi bolsista de iniciação científica CNPQ, trabalhando em projetos relacionados à metaheurísticas. Além disso, participou de projetos relacionados à processamento de imagens e reconhecimento de padrões. Continuou os estudos no programa de Mestrado em Informática da Pontifícia Universidade Católica do Rio de Janeiro / PUC-Rio. Durante o mestrado, atuou em projetos voltados à indústria do petróleo no laboratório Tecgraf.

Ficha Catalográfica

Araújo, Chrystiano Barbosa de Souza

Geração de malha de elementos finitos com inserção de restrições a posteriori / Chrystiano Barbosa de Souza Araújo; orientador: Waldemar Celes Filho. — 2012.

52 f: il. (color.); 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2012.

Inclui bibliografia.

1. Informática – Teses. 2. Geração automática de malhas. 3. Método de elementos finitos. 4. Refinamento. I. Celes Filho, Waldemar. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Agradecimentos

Agradeço primeiramente a Deus. Sem ele, jamais teria conseguido chegar até aqui. Ainda há muito que percorrer, mas tenho certeza de que sua companhia me fortalece para continuar caminhando.

Agradeço ao meu orientador Waldemar pela confiança e atenção depositada na realização deste trabalho, os quais foram de extrema importância na manutenção da motivação e perseverança necessária para o desenvolvimento de um trabalho de pesquisa.

Agradeço a todas as pessoas que fazem parte da minha grande e amada família. Obrigado pela compreensão quando precisei estar ausente em datas mais que especiais. Aos meus avós Célia e Zezinho pelo maior exemplo de superação e determinação que poderia ter. Aos meus pais Paulo e Tania por todo o apoio, por ter acreditado em meu potencial mesmo quando eu duvidei. Aos meus irmãos Thiago, Ricardo, Gustavo e Guilherme, pela cumplicidade e exemplo de amizade em todas as horas. A minha namorada Samyra por estar presente em minha vida mesmo nos momentos mais difíceis.

Agradeço a todos os novos amigos criados durante o mestrado. Em especial, agradeço ao Thiago, Marcelo, Allan, José, Renato, Fabrício, Alessandro, Fabio e Eduardo, os quais estiveram mais próximos no período de realização deste trabalho. Também gostaria de agradecer ao Rodrigo Espinha pela contribuição na realização desta pesquisa.

Aos professores Ítalo Matias e Fabricio Barros por acreditar em mim desde meus primeiros passos na Computação.

Agradeço ao laboratório Tecgraf pela oportunidade de fazer parte desse grupo e por proporcionar um ambiente diferenciado para a realização de trabalhos de pesquisa.

Meus sinceros agradecimentos a todas as pessoas que estiverem comigo nessa grande jornada.

Resumo

Araújo, Chrystiano Barbosa de Souza; Celes Filho, Waldemar. **Geração de malha de elementos finitos com inserção de restrições a posteriori**. Rio de Janeiro, 2012. 52p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

O método de elementos finitos é uma ferramenta amplamente utilizada na análise e simulação de fenômenos físicos. Uma etapa crucial desse método consiste na discretização de um domínio contínuo em uma malha de elementos finitos. A precisão da solução obtida na análise está diretamente relacionada à qualidade da malha utilizada, a qual é determinada pela forma e consistência de seus elementos. Grande parte dos trabalhos existentes na literatura é baseada em métodos clássicos, como Avanço de Fronteira, Triangulação de Delaunay com Restrição e Decomposição de Domínios. Utilizando uma abordagem diferente, esta dissertação propõe um método de geração a posteriori, em que restrições do domínio são inseridas iterativamente na malha, a qual é deformada localmente a fim de satisfazer cada nova restrição. Para validar o método proposto, os resultados obtidos são analisados através de critérios de qualidade comumente utilizados na literatura.

Palavras-chave

Geração automática de malhas ; Método de elementos finitos ; Refinamento.

Abstract

Araújo, Chrystiano Barbosa de Souza; Celes Filho, Waldemar (advisor). **Finite element mesh generation with deferred constraint insertion**. Rio de Janeiro, 2012. 52p. MSc. Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The finite element method (FEM) is a widely used tool in the analysis and simulation of physical phenomena. The discretization of a continuous domain into a finite element mesh is an essential FEM step. The accuracy of the solution is highly dependent on the quality of the input mesh, which is mainly evaluated through the shape of the elements. Most of the related works is based on traditional methods, such as Advancing Front, Constrained Delaunay Triangulation and Domain Decomposition. Unlike these methods, this work proposes an iterative mesh generation method with deferred constraint insertion, in which an initially regular mesh is locally deformed in order to satisfy each new domain constraint. In addition, in order to validate the proposed method, this work evaluates each output mesh according to quality criteria commonly used in the literature.

Keywords

Automatic mesh generation; Finite element method; Mesh Refinement.

Sumário

| | | |
|-----|---|-----------|
| 1 | Introdução | 9 |
| 2 | Trabalhos Relacionados | 12 |
| 2.1 | Métodos diretos | 13 |
| 2.2 | Métodos Indiretos | 15 |
| 3 | Método Proposto | 18 |
| 3.1 | Geração da malha de triângulos | 19 |
| 3.2 | Conversão em uma malha de quadriláteros | 29 |
| 4 | Resultados | 36 |
| 4.1 | Triangulação de domínios complexos | 36 |
| 4.2 | Robustez do método de triangulação | 40 |
| 4.3 | Conversão em uma malha de quadriláteros | 42 |
| 5 | Conclusão | 48 |
| 6 | Referências Bibliográficas | 50 |

Para realizar grandes conquistas, devemos não apenas agir, mas também sonhar; não apenas planejar, mas também acreditar.

Anatole France

1

Introdução

O Método de Elementos Finitos (MEF) consiste em um método numérico amplamente utilizado na análise de diversos fenômenos físicos, como transferência de calor, deformações, mecânica dos fluidos, etc. A análise através do MEF é feita através de uma solução numérica aproximada do problema que ocorre em um domínio contínuo. Para isso, uma das etapas que compõe tal método consiste na discretização do domínio de interesse em um conjunto finito de nós e elementos. No caso de domínios planares, o domínio é geralmente discretizado em um conjunto de elementos triangulares ou quadrilaterais. O conjunto dos elementos e nós gerados é chamado de malha de elementos finitos. O erro obtido na aproximação da solução depende de inúmeros fatores, entre os quais pode-se destacar a natureza do problema, a qualidade e quantidade dos elementos e a posição dos nós amostrados. Com isso, a malha de elementos finitos utilizada na análise do problema possui total influência na qualidade da solução obtida.

O domínio do problema a ser analisado é descrito através restrições de fronteira e no seu interior. As restrições de fronteira consistem na divisão entre regiões do espaço discretizadas e não discretizadas, as quais são comumente utilizadas para representar os limites e buracos do domínio em questão. As restrições no interior são utilizadas para separar regiões do espaço com diferença significativa na propriedade física representada ou para representar descontinuidades. Em uma malha de elementos finitos 2D, ambos os tipos de restrição devem ser representados por um conjunto de arestas na malha.

Além da conformidade com a geometria do domínio, uma malha adequada para a análise de elementos finitos deve ser válida do ponto de vista topológico e geométrico. Para isso, a interseção entre os elementos que a compõem deve existir apenas nos seus vértices ou arestas. A malha é considerada geometricamente consistente caso nenhum de seus elementos possua área zero. Além das condições citadas, uma malha de elementos finitos deve ser composta de elementos considerados de qualidade. Diante disso, a malha resultante da discretização deve ser conforme ao domínio de entrada, consistente do ponto de vista topológico e geométrico, assim como composta de elementos de quali-

dade. Tais condições tornam a geração automática de malhas como uma etapa crucial no método de elementos finitos.

O problema de triangulação de um determinado domínio geométrico é uma área relativamente consolidada, a qual é dominada principalmente por métodos baseados na Triangulação de Delaunay e Avanço de Fronteiras. Contudo, de acordo com Liu et al. (12), a malha de quadriláteros é mais adequada para uma análise realizada com o método de elementos finitos. Diferente da triangulação, a geração automática de malhas de quadriláteros ainda é uma área com problemas em aberto, especialmente quando aplicada a domínios complexos (10).

Os algoritmos de geração de malhas de quadriláteros são classificados em métodos indiretos e diretos. Os métodos indiretos utilizam algum algoritmo existente na literatura para discretizar o domínio de entrada em uma malha de triângulos. Posteriormente, essa triangulação é convertida em uma malha composta de elementos quadrilaterais. Nos métodos diretos, o domínio de entrada é diretamente discretizado em uma malha de quadriláteros. Os métodos diretos geralmente apresentam soluções de maior qualidade, sobretudo nas regiões próximas às fronteiras do domínio. Porém, a aplicação de tais métodos é restrita a domínios mais simples.

Em geral, os métodos de geração de malhas de quadriláteros existentes na literatura apresentam alguma limitação em relação ao domínio geométrico ao qual são aplicados. Seja em relação a sua complexidade ou até mesmo em relação aos tipos de restrição que o compõem. Além disso, os métodos classificados como indiretos geralmente utilizam alguma variação da Triangulação de Delaunay para realizar a primeira etapa do método. Contudo, os métodos baseados na Triangulação de Delaunay dependem de um conjunto de pontos distribuídos sobre o domínio em questão, o qual pode ser difícil de ser obtido em domínios complexos. A criação de um método de geração de malhas de quadriláteros independente da complexidade e tipos de restrição do domínio motivou o desenvolvimento desta dissertação.

Este trabalho propõe um novo método de geração de malhas de quadriláteros para domínios planares. O método proposto é classificado como indireto e, portanto, divide-se em duas principais etapas. Na primeira delas, o domínio de entrada é discretizado em uma malha de triângulos através de um processo adaptativo. Para isso, uma malha inicial é definida sobre o domínio, a qual é iterativamente modificada para satisfazer cada restrição inserida. A malha de triângulos gerada é posteriormente convertida em uma malha composta apenas de elementos quadrilaterais. O método de conversão proposto neste trabalho é inspirado no algoritmo apresentado por Velho (21), o qual combina

técnicas de pareamento e subdivisão dos triângulos para gerar os quadriláteros da malha final.

Testes realizados demonstram que ambos os algoritmos propostos nesta dissertação, geração da malha de triângulos e conversão da malha para quadriláteros, apresentam resultados com qualidade mesmo quando aplicados a domínios complexos. Além disso, o método proposto é automático, partindo de uma malha inicial simples, e muito eficiente. Diante disso, este trabalho apresenta contribuições em ambas as etapas que compõem o método indireto proposto.

Esta dissertação está organizada da seguinte forma: o Capítulo 2 apresenta uma breve descrição dos trabalhos existentes para geração de malhas de quadriláteros em domínios planares. O método proposto é detalhado no Capítulo 3. Os resultados obtidos são apresentados no Capítulo 4. Por fim, o Capítulo 5 conclui este trabalho.

2 Trabalhos Relacionados

Grande parte dos trabalhos existentes na literatura para a geração de malhas de quadriláteros para domínios planares são aplicados a domínios compostos apenas de restrições do tipo *closed boundary* (10), as quais são utilizadas para definir as fronteiras e buracos no interior do domínio (Figura 2.1). Contudo, restrições do tipo *open boundary*, Figura 2.1, são especialmente importantes para representar falhas e horizontes em modelos geológicos (12) (Figura 2.2); assim como na modelagem de estruturas navais e aeronáuticas, nas quais reforços são representados como restrições de linhas em domínios 2D (10)(11).

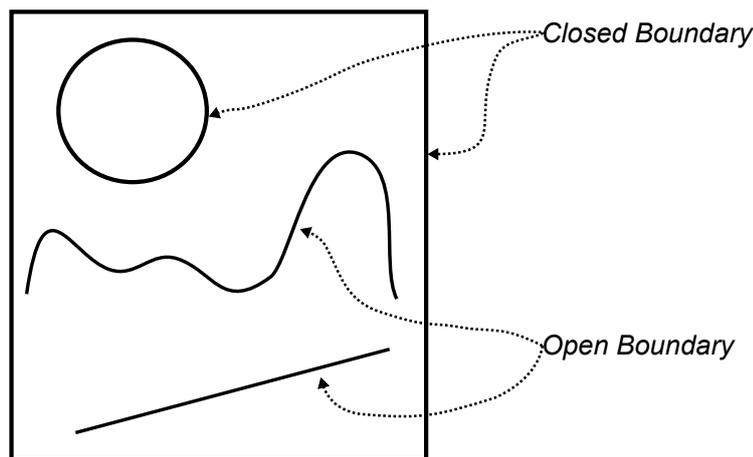


Figura 2.1: Restrições *closed* e *open boundary*.

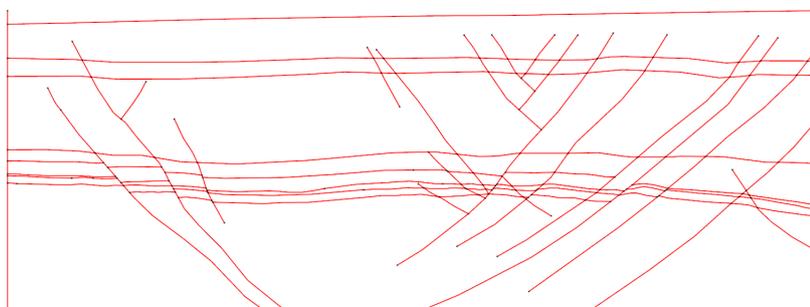


Figura 2.2: Exemplo de modelo geológico.

Os métodos de geração de malhas de quadriláteros podem ser classificados em dois grupos principais: métodos diretos e indiretos (16). Nos métodos classificados como indiretos, o domínio geométrico é inicialmente discretizado em uma malha intermediária de triângulos, a qual é posteriormente convertida em uma malha de elementos quadriláteros através da combinação ou subdivisão dos triângulos existentes. Por outro lado, nos métodos classificados como diretos, o domínio de entrada é discretizado diretamente em uma malha de quadriláteros sem a necessidade da etapa de triangulação.

Os algoritmos classificados como diretos geralmente apresentam resultados com qualidade superior aos obtidos por métodos indiretos, sobretudo nas regiões próximas às restrições do domínio (10); entretanto, tais algoritmos costumam ser ineficientes quando aplicados a domínios complexos, visto que em casos passíveis de solução, os resultados obtidos apresentam baixa qualidade. Em contrapartida, os métodos indiretos são capazes de obter soluções com qualidade satisfatória mesmo para domínios complexos (12).

2.1

Métodos diretos

Os métodos diretos são geralmente baseados em duas principais técnicas. A primeira delas engloba métodos em que o domínio geométrico de entrada é decomposto em regiões mais simples, chamada de Decomposição de Domínios. Devido à simplicidade das regiões decompostas, tais regiões são discretizadas utilizando métodos tradicionais, e a solução global é obtida através da combinação dos subproblemas resolvidos localmente. O principal desafio dos algoritmos baseados nessa técnica reside na escolha da partição adequada do domínio. Ao longo dos anos, diversas formas para realizar essa decomposição foram propostas. Um dos primeiros métodos a aplicar essa técnica na geração de malhas de quadriláteros foi proposta por Baehmann et al. (1), a partir da qual o domínio é decomposto recursivamente utilizando a estrutura *quadtree*. Cada subregião gerada é discretizada em um conjunto de quadriláteros, e os vértices gerados são reposicionados para melhor aproximar o domínio geométrico. No método proposto por Tam e Armstrong (20), um domínio é decomposto utilizando informações do seu eixo medial. O algoritmo proposto por Nowotny (15) utiliza um método de otimização geométrica para definir a melhor forma de realizar a decomposição em sub-regiões, as quais são discretizadas em uma malha de quadriláteros utilizando a técnica de Avanço de Fronteiras. Recentemente, Lin et al. (11) propuseram um método baseado em Decomposição de Domínios para tratar domínios com restrição de linha. Basicamente, as sub-regiões são geradas de maneira que as restrições de linha existentes no domínio

inicial se tornem restrições de fronteiras nas regiões geradas. A partir disso, métodos tradicionais podem ser aplicados para discretização das regiões geradas. Um exemplo da decomposição de um domínio em sub-regiões pode ser observado na Figura 2.3, cujas fronteiras entre as partições estão discretizadas de forma uniforme para facilitar a etapa de combinação das regiões.

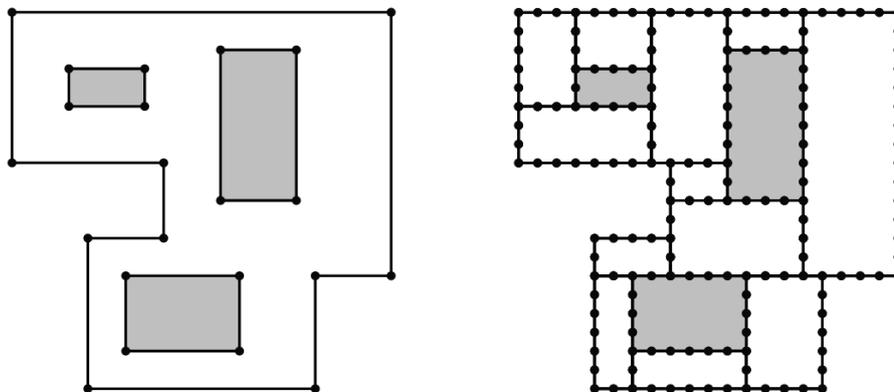


Figura 2.3: Exemplo do resultado da técnica de Decomposição de Domínios (15).

Algoritmos classificados nessa categoria são eficientes na geração de malhas de boa qualidade; porém, a automatização de tais métodos se torna difícil em razão da quantidade de parâmetros utilizados, sobretudo quando aplicados a domínios complexos (10).

A segunda categoria engloba métodos baseados na técnica de Avanço de Fronteiras. Basicamente, esses métodos iniciam com a definição de um conjunto de arestas chamado fronteira ativa, o qual é inicialmente definido como o resultado da discretização das curvas do domínio. O tamanho definido na amostragem da curva reflete no tamanho inicial dos elementos gerados. O algoritmo seleciona progressivamente uma aresta no conjunto definido, a qual é utilizada para gerar um novo elemento através de sua projeção no interior do domínio. À medida que novos elementos são formados, a fronteira ativa é atualizada com as novas arestas criadas. O algoritmo termina quando todo o domínio for coberto por elementos. Seguindo essa linha, Blacker e Stephenson (3) apresentaram um método chamado de Paving, cuja geração dos elementos é realizada linha a linha. White e Kinney (25) apresentaram uma variação desse método, em que a formação é realizada elemento a elemento, ao contrário do método original. Mais recentemente, Park et al. (18) apresentaram outra variação desse método, tornando possível a sua aplicação em domínios com restrições *open boundary*.

Métodos nessa categoria geralmente apresentam resultados de qualidade, especialmente nas fronteiras do domínio. Contudo, a cada elemento criado,

existe a necessidade de detectar e solucionar possíveis interseções entre as arestas criadas, resultando na sobreposição entre os elementos criados (Figura 2.4(a)). Além disso, elementos com baixa qualidade são gerados em regiões que apresentam diferença significativa no tamanho dos elementos criados (10) (Figura 2.4(b)).

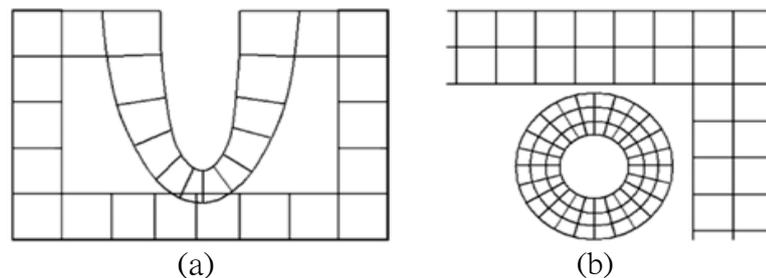


Figura 2.4: Limitações apresentadas por métodos baseados em avanço de fronteiras (10). (a) Sobreposição entre elementos. (b) Diferença significativa no tamanho de elementos criados a partir de fronteiras opostas.

2.2

Métodos Indiretos

Nos métodos classificados como indiretos, o domínio em questão é inicialmente discretizado em uma malha de triângulos por meio de algum método de triangulação existente na literatura. Em geral, essa etapa é realizada através de alguma variação da clássica Triangulação de Delaunay (10)(12).

A Triangulação de Delaunay de um conjunto de pontos no plano é a triangulação cuja soma dos ângulos mínimos de cada triângulo é máxima, resultando na melhor triangulação possível para o conjunto de pontos de entrada. Uma malha é considerada Delaunay se o circuncírculo de cada triângulo não possui nenhum dos pontos do conjunto de entrada em seu interior (16).

Apesar de garantir a triangulação com máxima soma dos ângulos mínimos, a Triangulação de Delaunay possui algumas desvantagens. Em aplicações envolvendo o método de elementos finitos, as restrições do domínio devem ser representadas através de um conjunto de arestas na malha gerada. Contudo, a Triangulação de Delaunay original leva em consideração apenas a distribuição dos pontos do conjunto de entrada, o que não garante a presença dos segmentos que definem a geometria do problema na malha de saída. Para solucionar tal problema, uma variação chamada de Triangulação de Delaunay com Restrições foi proposta por Chew (6). Após a discretização do domínio através da versão clássica da Triangulação de Delaunay, esse método consiste em identificar cada aresta definida como restrição e que não existe na malha

gerada. Cada uma delas é inserida na malha, e o domínio em torno de tal aresta é novamente triangulado (10). Deve-se ainda notar que a Triangulação de Delaunay tem como entrada uma nuvem de pontos, cuja geração pode ser uma tarefa difícil em caso de domínios complexos.

Na segunda e última etapa dos métodos indiretos, a malha de triângulos gerada é convertida em uma malha de quadriláteros. A forma mais simples de realizar essa conversão consiste na subdivisão de cada triângulo em três quadriláteros através da inserção de um novo vértice no baricentro de cada triângulo, e conectando-o com o ponto médio de cada uma das arestas, chamado de subdivisão de Catmull-Clark (4). Apesar de garantir uma malha *all-quad*, malha composta apenas de elementos quadrilaterais, a malha gerada possui baixa qualidade devido ao alto número de vértices irregulares, isto é, vértices internos cuja valência é diferente de quatro. Um exemplo dessa conversão pode ser observada na Figura 2.5.

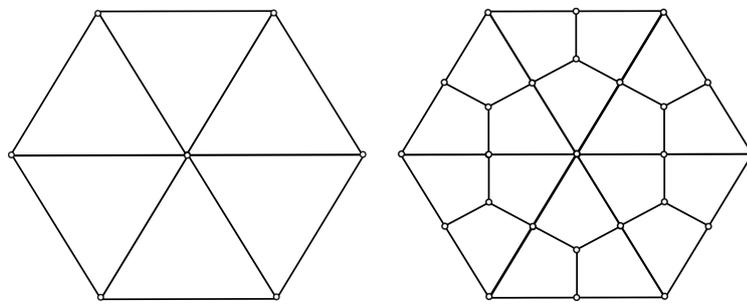


Figura 2.5: Conversão de uma malha de triângulos em quadriláteros através da subdivisão de Catmull-Clark.

No método proposto por Liu et al. (12), os triângulos são convertidos em quadriláteros utilizando o método proposto por Catmull-Clark. Para contornar os problemas inerentes a essa conversão, os quadriláteros resultantes são combinados utilizando um método baseado em avanço de fronteiras. Posteriormente, operadores topológicos e de suavização são utilizados para obter um resultado de maior qualidade.

O método Q-Morph, proposto por Owen et al. (17), baseia-se na técnica de avanço de fronteiras para realizar a geração dos quadriláteros através da combinação dos triângulos existentes na malha de entrada. Apesar de apresentar bons resultados, a aplicação de tal método limita-se a domínios contendo apenas restrições *closed boundary*. Além disso, esse método não garante a preservação dos vértices existentes na malha de triângulos e, conseqüentemente, não garante a existência das restrições que definem a geometria do domínio (14). Lee et al. (10) apresentaram uma variação do método Q-Morph, tornando possível a sua utilização em domínios com restrições de linha.

O método Q-Tran, proposto em (14), inicia com a classificação de cada aresta existente na malha de triângulos de entrada. Cada par de triângulos adjacentes é convertido em quadriláteros utilizando uma estratégia de acordo com a classificação da aresta compartilhada.

O método proposto por Velho (21) utiliza a estrutura de malhas 4-8 para realizar a conversão em quadriláteros. Esse método inicia com o pareamento de triângulos adjacentes (*cluster*) e identificação dos triângulos isolados, os quais não fazem parte de nenhum *cluster* formado. Posteriormente, os *clusters* são convertidos em um conjunto de quatro quadriláteros, assim como os triângulos isolados em três quadriláteros, garantindo uma malha de saída composta apenas de elementos quadriláteros.

3

Método Proposto

O método de geração de malhas de quadriláteros proposto neste trabalho não faz qualquer distinção entre as restrições existentes no domínio, podendo ser aplicado a domínios planares definidos por qualquer tipo de restrição: *open boundary* e *closed boundary*. O domínio geométrico de entrada D é descrito através de um conjunto de m curvas poligonais $D = \{C_1, C_2, \dots, C_m\}$. Cada curva $C_i \in D$ é definida por uma sequência finita $(p_0, e_0, p_1, e_1, \dots, e_{n-2}, p_{n-1})$, onde p_0, \dots, p_{n-1} são pontos do \mathbb{R}^2 e e_i consiste em um segmento de reta com extremidades p_i, p_{i+1} ($i = 0, \dots, n - 2$). Toda curva $C \in D$ deve ser aberta, ou seja, o último e o primeiro ponto da curva devem ser diferentes, $p_0 \neq p_{n-1}$. Não deve existir auto-interseção, portanto, a interseção entre os segmentos e_i e e_{i+1} deve existir apenas no ponto p_{i+1} ($i = 0, \dots, n - 2$). Além disso, não deve existir interseção entre segmentos não consecutivos. Por fim, a interseção entre duas curvas distintas pode existir apenas nos pontos extremos das mesmas, $C_i \cap C_j = \{\}$ ou $\{p_0\}$ ou $\{p_{n-1}\}$ ou $\{p_0, p_{n-1}\}$.

Apesar da restrição no caso de curvas fechadas, essas podem ser facilmente utilizadas como entrada do método proposto dividindo-as em duas ou mais curvas abertas.

O algoritmo de geração proposto é classificado como um método indireto e, portanto, pode ser dividido em duas principais etapas. Na primeira delas, uma malha de triângulos é gerada para um dado domínio de entrada, a qual é posteriormente convertida em uma malha de elementos quadriláteros preservando as restrições existentes. O presente trabalho apresenta contribuições em ambas as etapas do método indireto, sendo portanto também um método válido e original para geração de malhas de triângulos.

Na primeira etapa, este trabalho propõe um algoritmo adaptativo para geração de uma malha de triângulos a partir de um domínio planar arbitrário. O algoritmo inicia com a definição de uma malha de triângulos envolvendo todo o domínio de entrada. Através de um processo iterativo, a malha em questão é localmente modificada para se adaptar a cada curva do domínio. A inserção de uma determinada curva é realizada através de um caminhamento entre dois vértices da malha, os quais são escolhidos (ou criados) mediante a

um critério de qualidade e posteriormente projetados nos extremos da curva em questão. A cada passo do caminhamento, um novo vértice é escolhido, inserido no caminho e também projetado na curva. A escolha de cada vértice deve satisfazer algumas restrições impostas, as quais garantem a qualidade e consistência da malha durante todo o processo de adaptação. Além disso, a utilização de tais restrições combinada ao uso de operadores de refinamento garante a convergência do algoritmo independente da resolução da malha inicial. Sendo possível, mesmo para uma malha inicial grosseira, obter uma malha de triângulos de qualidade para um dado domínio complexo.

Na segunda e última etapa, o método de conversão da malha gerada na etapa anterior em uma malha de quadriláteros é inspirado no algoritmo proposto por Velho (21), o qual consiste em um método de conversão local que utiliza a estrutura de malhas 4-8 para realizar o pareamento dos triângulos. Basicamente, o algoritmo consiste em classificar os triângulos da malha em dois conjuntos: conjunto de *clusters*, formado através do pareamento de triângulos adjacentes; e o conjunto dos triângulos isolados, formado pelos triângulos que não pertencem a nenhum *cluster* criado. Em seguida, os triângulos são subdivididos de acordo com uma estratégia predefinida para gerar os quadriláteros da malha final.

O método proposto nessa dissertação foi implementado utilizando a biblioteca TopS (5), a qual fornece uma estrutura de dados topológica para representação de malhas.

Cada uma das etapas que compõem o método de geração proposto será descrita nas próximas seções.

3.1

Geração da malha de triângulos

O método de geração de malhas de triângulos proposto consiste em um algoritmo adaptativo, o qual realiza modificações locais em uma malha existente para adaptá-la a cada restrição inserida. A primeira etapa deste método consiste na definição de uma malha inicial M envolvendo todo o domínio geométrico de entrada. Posteriormente, cada restrição $C \in D$ será inserida iterativamente. A inserção de uma restrição C inicia com a atração de dois vértices, x_{source} e x_{target} , aos pontos extremos p_0 e p_{n-1} de C . A próxima etapa pode ser compreendida como um caminhamento entre os vértices x_{source} e x_{target} , cuja escolha de cada novo vértice inserido no caminho é realizada utilizando um critério de distorção descrito nessa seção. Cada novo vértice inserido é imediatamente atraído a um ponto na curva, o que resulta na mudança da qualidade dos triângulos na estrela deste vértice. Neste trabalho,

a estrela de um vértice x_i é denotada por $N_1(x_i)$, a qual consiste no conjunto de elementos diretamente adjacentes à x_i . Para garantir a qualidade da malha durante todo o processo de adaptação, um conjunto de operadores de melhoria é aplicado localmente nos elementos pertencentes à estrela do vértice atraído. Uma visão geral do método é apresentada no Algoritmo 1, e cada uma de suas etapas é detalhada nas próximas seções.

Algoritmo 1: Geração da malha de triângulos para um dado domínio geométrico.

```

1 Definição da malha de triângulos inicial  $M$ ;
2 for  $C \in D$  do
3   Atrair dois vértices  $(x_{source}, x_{target})$  aos pontos  $p_0, p_{n-1} \in C$ ;
4   Aplicar operadores de melhoria na adjacência de  $(x_{source}$  e  $x_{target})$ ;
5    $x_{last} \leftarrow x_{source}$ ;
6   while  $x_{last} \neq x_{target}$  do
7     Definir vértice  $x_{best} \in N_1(x_{last})$  para fazer parte do caminho;
8     Atrair vértice  $x_{best}$  em  $C$  ;
9     Aplicar operadores de melhoria na adjacência de  $x_{best}$ ;
10     $x_{last} \leftarrow x_{best}$ ;
11  end
12 end

```

3.1.1

Definição da malha inicial

A primeira etapa do método consiste na definição de uma malha de triângulos envolvendo todo o domínio de entrada. O algoritmo proposto presume apenas que a malha inicial não possui elementos degenerados ou invertidos. Nenhuma condição em relação à resolução ou qualidade da mesma é imposta. Sendo assim, o método proposto é capaz gerar uma malha de triângulos de qualidade para domínios complexos, independente da resolução ou estrutura da malha inicial. Contudo, neste trabalho, a malha inicial é definida por uma malha 4-8 devido às propriedades inerentes a sua conectividade. A estrutura topológica deste tipo de malha também é explorada na conversão em uma malha de quadriláteros.

Uma malha é dita 4-8 quando sua conectividade é equivalente a *tilings* ou partições $[4.8^2]$, chamadas de *Laves Tilings*, cujas partições de um determinado plano são definidas através de triângulos retângulos isósceles, resultando em uma malha composta de vértices com valência 4 e 8 (24), como ilustrado na Figura 3.1. A estrutura básica desse tipo de *tiling* é composta por um par de

triângulos que define um bloco quadrado dividido por uma de suas diagonais, chamado de bloco básico (24). A malha definida no início do processo de adaptação é formada por *tilings* 4-8 em razão da estrutura básica desse tipo de malha, quadrangulação triangulada (23), a partir da qual os quadriláteros serão formados na última etapa do método.

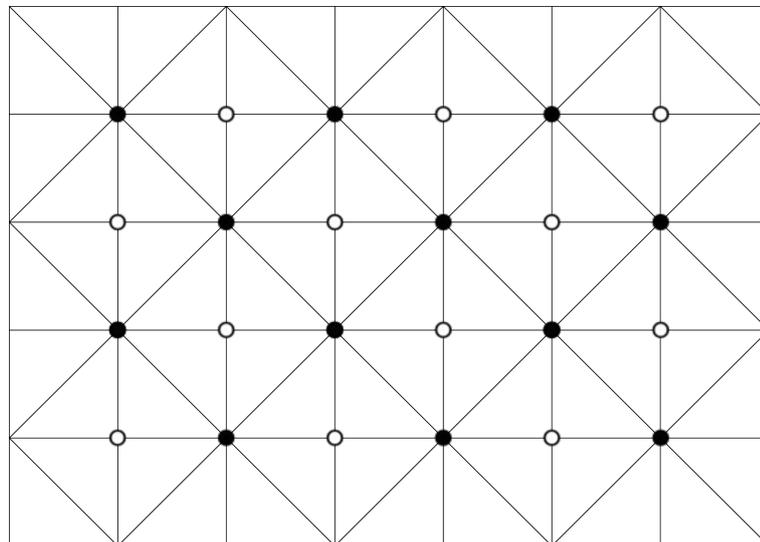


Figura 3.1: Estrutura de uma malha 4-8. Vértices com valência 4 são destacados por círculos brancos, enquanto os vértices com valência 8 são destacados por círculos pretos.

O operador de refinamento utilizado em malhas 4-8 consiste na subdivisão por *bisseção*, em que um bloco básico é dividido em quatro triângulos através da operação de *bisseção* na sua aresta interna, ilustrado na Figura 3.2. O refinamento por *bisseção* de um bloco básico inicia com a inserção de um novo vértice no ponto médio da aresta interna; posteriormente, cada triângulo que compõe o bloco será dividido em dois novos através da criação de uma nova aresta conectando o novo vértice ao vértice oposto da face dividida (22).

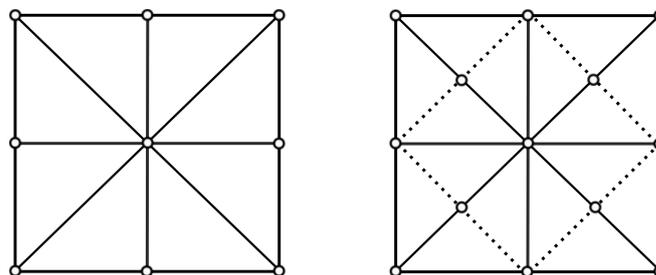


Figura 3.2: Refinamento 4-8.

A utilização da *bisseção* como operador de refinamento em uma malha 4-8 garante a preservação das propriedades da estrutura. Contudo, à medida que os vértices são reposicionados para representar as restrições inseridas,

a aplicação de tal operador pode apresentar uma característica indesejada. Esta característica ocorre quando a aresta interna do bloco básico, aresta subdividida pelo operador, é significativamente menor que as demais arestas do bloco, Figura 3.3(a). A utilização deste operador de subdivisão na aresta interna do bloco resultaria na geração de novos triângulos com baixa qualidade, Figura 3.3(b). Diante disso, esse trabalho utiliza um operador de refinamento chamado de *Longest Edge Bisection*. Neste caso, um triângulo é sempre subdividido em dois através da *bisseção* na maior aresta. Os triângulos gerados por esse operador possuem qualidade superior aos gerados pela *bisseção* na aresta interna, como pode ser observado na Figura 3.3(c).

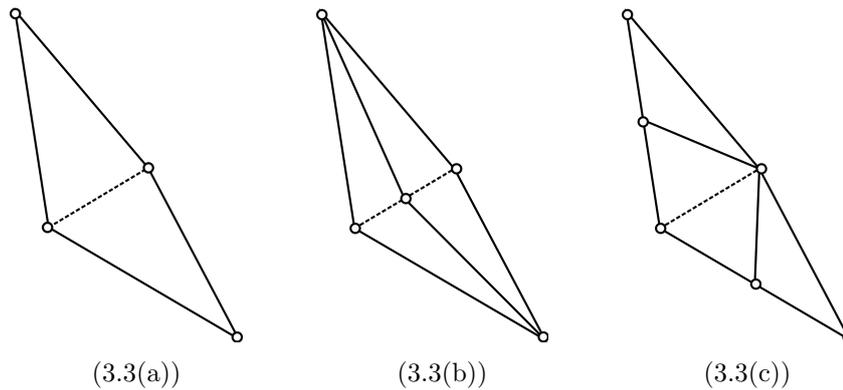


Figura 3.3: Subdivisão pela maior aresta de um triângulo.

A aplicação da subdivisão pela maior aresta de um triângulo não preserva as propriedades de uma malha 4-8, podendo resultar em vértices com valência superior a 8. Contudo, segundo (19), a quantidade de blocos básicos, cuja maior aresta é a interna, e a qualidade dos triângulos gerados aumenta à medida que o refinamento pela maior aresta é utilizado, o que torna esse operador adequado para o método proposto.

3.1.2

Inserção de uma restrição

A inserção de cada restrição é realizada através de modificações locais na malha M para representar a nova restrição inserida. A inserção de uma determinada curva $C \in D$ inicia com a atração de dois vértices x_{source} e x_{target} aos pontos extremos $p_0, p_{i-1} \in C$. O conjunto de arestas que representará a restrição inserida é definido através de um caminhamento entre os dois vértices x_{source} e x_{target} . A cada passo do caminhamento, um novo vértice é inserido no caminho percorrido e imediatamente atraído a um determinado ponto da curva. Para reduzir a distorção causada pela movimentação de cada vértice atraído, operadores locais de melhoria são utilizados após a atração de cada

vértice. O processo de adaptação termina quando o vértice x_{target} fizer parte do caminho percorrido.

A escolha do vértice atraído no início e durante o caminhamento é realizada por meio da avaliação da distorção causada por sua atração à curva, a qual é definida como a máxima redução de qualidade dos triângulos na estrela do vértice em questão. Dessa forma, a distorção provocada pela atração de um vértice x_i a um determinado ponto $p \in C$ é definida pela Equação 3-1.

$$d(x_i, p) = \max_{j \in N_1(x_i)} \left(\frac{\phi(t_j) - \phi(t_j')}{\phi(t_j)} \right) \quad (3-1)$$

onde $\phi(t_j)$ e $\phi(t_j')$ definem a qualidade do triângulo j , na estrela de x_i , antes e depois da atração do vértice x_i ao ponto p , respectivamente. Neste trabalho, a qualidade de um triângulo é definida através do parâmetro de Lo (13), o qual é definido pela Equação 3-2.

$$\phi(t) = 4\sqrt{3} \frac{A}{\sum_{i=1}^3 l_i^2} \quad (3-2)$$

onde A consiste na área do triângulo t e l representa o tamanho de cada lado. O parâmetro de Lo varia entre 0 (triângulos com área zero) e 1 (triângulos equiláteros).

O conjunto K de vértices candidatos à atração a um dos pontos extremos p_e de C é definido pelos vértices que compõem o triângulo cujo ponto p_e é interior. O melhor dos candidatos $x_{best} \in K$ é definido pelo vértice cuja distorção causada por sua atração é mínima, ou seja, $x_{best} = \{x_i \in K | d(x_i, p_e) = \min_{x_j \in K} (d(x_j, p_e))\}$. A atração de um vértice a qualquer ponto de uma curva tem influência na qualidade de todos os triângulos na sua estrela, o que pode resultar em triângulos degenerados ou até mesmo invertidos. Para garantir a consistência da malha durante o processo de adaptação, a atração de um determinado vértice candidato é realizada mediante algumas restrições:

- A qualidade mínima dos triângulos na estrela do vértice candidato x_i , levando em consideração sua nova posição, deve ser superior a uma constante ϕ_{min} , Equação 3-3. Nos testes realizados, a constante ϕ_{min} foi definida como 0,1.

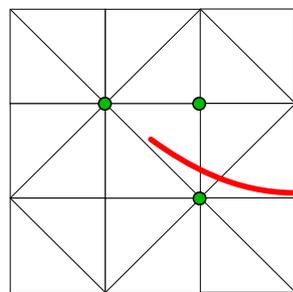
$$\min_{j \in N_1(x_i)} (\phi(t_j')) > \phi_{min} \quad (3-3)$$

- A redução máxima da qualidade dos elementos na estrela do vértice candidato x_i deve ser inferior a d_{max} , Equação 3-4. Nos testes realizados, d_{max} foi definido como 0,7.

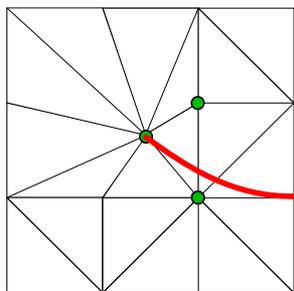
$$d(x_i, p) < d_{max} \quad (3-4)$$

- Caso x_i seja um vértice fixo em alguma outra curva, a atração de x_i será válida se a distorção provocada for nula, $d(x_i, p) = 0$.

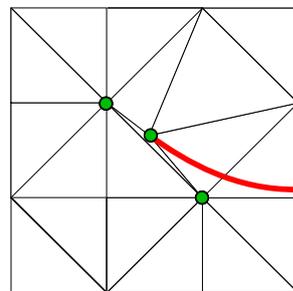
Caso o vértice definido como melhor candidato satisfaça as condições de atração descritas, este será atraído e definido como um vértice fixo na curva em questão. Caso contrário, o triângulo cujo ponto extremo é interior será subdividido pela sua maior aresta e o processo refeito até que um candidato válido seja encontrado. Na Figura 3.4(a), os vértices candidatos à atração ao ponto extremo da curva (vermelho) são representados em verde. A atração de cada um dos candidatos é ilustrada nas Figuras 3.4(b), (c) e (d). Observe-se que o resultado apresentado na Figura 3.4(b) possui a menor redução de qualidade dos triângulos na estrela do vértice atraído, resultando no melhor candidato à atração.



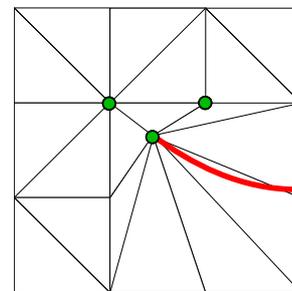
(3.4(a))



(3.4(b))



(3.4(c))



(3.4(d))

Figura 3.4: Atração do vértice ao ponto extremo da curva.

Além da necessidade de uma malha consistente do ponto de vista geométrico e topológico, propriedades essas garantidas com as restrições de atração descritas, uma malha adequada para análise deve também possuir elementos de qualidade. Com este objetivo, uma combinação de métodos de melhoria e suavização é utilizada em uma etapa posterior à atração de cada vértice, garantindo dessa forma a qualidade da malha durante todo o processo de adaptação. Como pode ser observada na Figura 3.4, a movimentação de um determinado vértice tem influência na qualidade apenas dos triângulos na sua

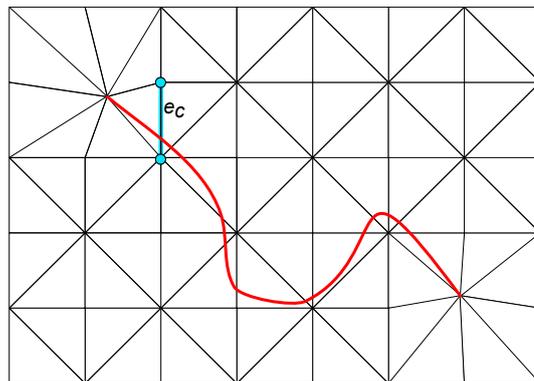


Figura 3.5: Definição dos nós candidatos à atração.

estrela. Diante disso, a etapa de melhoria é composta apenas de operadores locais, os quais são descritos na Seção 3.1.3.

Após a escolha e atração dos vértices x_{source} e x_{target} , a próxima etapa do algoritmo consiste no caminhamento entre esses dois vértices. O último vértice alcançado por uma determinada iteração do caminhamento é chamado de x_{last} , o qual inicialmente é definido como x_{source} . Similar à atração dos vértices aos extremos da curva, cada passo do caminhamento pode ser dividido em três principais etapas: (1) definição do próximo vértice a fazer parte do caminho; (2) validação e possível atração do vértice escolhido; (3) melhoria da qualidade na estrela do vértice atraído.

No início de cada iteração, passo (1), um vértice adjacente à x_{last} será escolhido como candidato ao próximo vértice a ser inserido no caminho percorrido. Independente da valência de x_{last} , apenas dois de seus vizinhos serão considerados vértices candidatos. Para definir tais vértices, encontra-se a aresta oposta ao vértice x_{last} intersectada pela curva C , chamada de aresta e_c . A partir disso, o conjunto de vértices candidatos à atração K é definido pelos vértices da aresta e_c , Figura 3.5. A atração de um vértice diferente dos definidos no conjunto K resultaria em triângulos degenerados ou invertidos e, por esse razão, são desconsiderados na escolha do próximo vértice do caminho. O vértice $x_i \in K$, cuja atração no ponto de interseção entre e_c e C resulta na menor distorção em sua estrela, será definido como o melhor dos candidatos x_{best} , semelhante à escolha dos vértices x_{source} e x_{target} . Caso x_{best} seja um vértice fixo a uma curva e diferente de x_{target} , o outro vértice do conjunto K será considerado como a melhor opção à atração. Caso os dois vértices do conjunto K sejam fixos e diferentes de x_{target} , a aresta e_c será refinada e o passo corrente do caminhamento feito.

Existem casos em que mais de uma aresta oposta ao vértice x_{last} é intersectada pela curva. Nesses casos, a aresta e_c será definida como a aresta com a primeira interseção na curva. Um exemplo deste caso é ilustrado na

Figura 3.6, em que todas as arestas intersectadas pela curva e opostas ao vértice x_{last} são destacadas em verde.

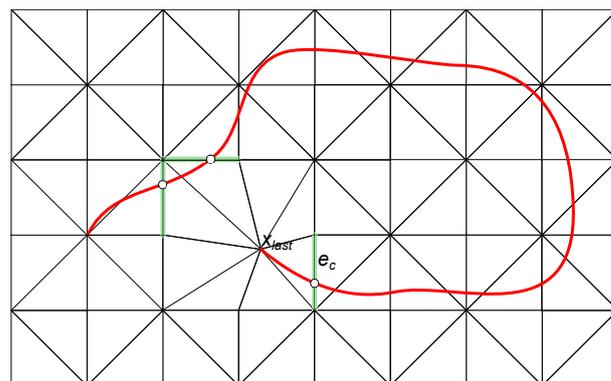


Figura 3.6: Definição da aresta e_c .

No passo (2), a atração do vértice x_{best} é validada. Assim como na validação da atração dos vértices x_{source} e x_{target} , a atração de x_{best} deve satisfazer as Equações 3-3 e 3-4; caso contrário, a aresta e_c será refinada e o passo corrente do caminhamento refeito. Além disso, caso a atração não satisfaça a condição 3-4, a tolerância de redução de qualidade d_{max} será relaxada antes do reinício do passo corrente.

Além das condições citadas, a possível aproximação da curva C pela aresta $e = \langle x_{last}, x_{best} \rangle$ deve apresentar um erro de aproximação δ inferior a uma determinada tolerância δ_{max} , $\delta(e, C) \leq \delta_{max}$. Onde $\delta(e, C)$ é definido como a distância perpendicular do ponto médio da aresta e até a curva C , Figura 3.7. Caso a aproximação não satisfaça essa condição, a aresta e será refinada e o passo corrente do caminhamento será refeito.

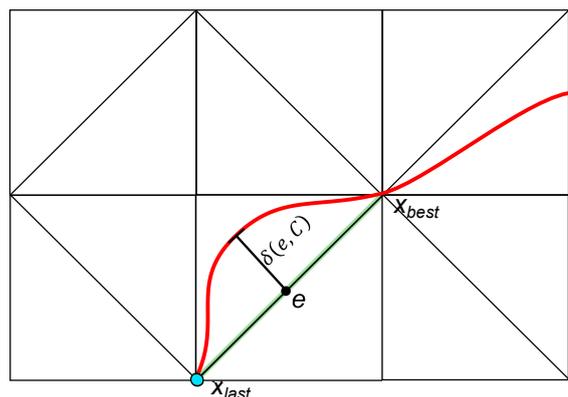


Figura 3.7: Erro de aproximação.

Após a escolha de um vértice $x_{best} \in N_1(x_{last})$ que satisfaça todas as condições de atração, o vértice escolhido será atraído ao ponto de interseção entre a curva C e a aresta e_c . Assim como na atração dos vértices x_{source} e

x_{target} , a atração de x_{best} é seguida da etapa de melhoria na qualidade dos elementos em sua estrela, a qual é descrita na Seção 3.1.3.

A inserção de uma curva termina quando o último vértice alcançado pelo caminhamento for o vértice x_{target} .

3.1.3

Operadores de melhoria de qualidade

A etapa de melhoria da qualidade dos elementos da malha é composta pela combinação de dois operadores amplamente utilizados na literatura, Laplaciano e *Flipping* de arestas. O Laplaciano consiste em um operador de suavização e, portanto, a qualidade dos elementos da malha é maximizada através do reposicionamento dos vértices existentes, preservando a topologia da mesma. Seguindo uma linha diferente, o *Flipping* de arestas realiza mudanças nas relações de adjacência da malha para obter a melhoria nos triângulos envolvidos. Ambos os operadores utilizados serão detalhados a seguir.

Operador Laplaciano

O operador Laplaciano é o método de suavização mais utilizado na literatura devido a sua simplicidade e eficiência (2). Nesse operador, cada vértice é iterativamente movido em direção ao baricentro do polígono formado pelos vértices na sua estrela. A cada iteração, uma nova posição x'_i de um determinado vértice x_i é calculada por $x'_i = x_i + \lambda L(x_i)$, em que λ consiste na velocidade da movimentação e $L(x_i)$, chamado de Laplaciano, consiste no vetor que define a direção da movimentação. Na versão mais simples, chamada de *Umbrella Operator*, o vetor Laplaciano pode ser calculado pela equação 3-5.

$$L(x_i) = \frac{1}{\sum_{j \in N_1(x_i)} w_{ij}} \sum_{j \in N_1(x_i)} w_{ij} (x_j - x_i) \quad (3-5)$$

onde w_{ij} consiste no peso dado a cada vizinho $x_j \in N_1(x_i)$. Existem diversas variações do operador Laplaciano, as quais se diferem basicamente pelo cálculo do peso w_{ij} . Na versão mais simples, $w_{ij} = 1$ para todo $x_j \in N_1(x_i)$. Em outra versão, chamada *Scale Dependent Umbrella Operator*, o peso é calculado pela inversa da distância entre os dois vértices em questão, $w_{ij} = \|x_j - x_i\|^{-1}$ (7).

Como a suavização de um determinado vértice depende apenas da informação dos seus vizinhos diretos, e tal vizinhança não varia com o tamanho da malha, pode-se afirmar que o Laplaciano é um operador linear.

Caso um determinado vértice seja fixo a uma curva, sua nova posição será calculada normalmente pelo operador Laplaciano e posteriormente projetada na respectiva curva, exceto se o vértice em questão estiver fixo em um dos pontos extremos da curva. Nesse caso, sua movimentação será impedida.

Apesar de apresentar bons resultados na maioria dos casos, tal operador não garante a melhoria da qualidade de todos os elementos da malha, podendo resultar em triângulos com baixa qualidade ou até mesmo invertidos (8). Para solucionar tal problema, alguns métodos utilizam uma versão estendida chamada de *Constrained Laplacian Smoothing*, ou *Smart Laplacian Smoothing* [4]. Nessa versão, um vértice é movido na direção do Laplaciano se tal movimentação satisfizer uma determinada condição de qualidade. No presente trabalho, a suavização de um vértice é realizada se a qualidade mínima dos elementos na sua estrela, levando em consideração a nova posição, for superior a constante de qualidade mínima tolerável ϕ_{min} .

Flipping de arestas

O *Flipping* de arestas consiste em um operador topológico muito utilizado na melhoria da qualidade dos triângulos que compartilham uma determinada aresta, e também na redução da valência de um determinado vértice. Considere dois triângulos $t_0 = \langle x_0, x_1, x_2 \rangle$ e $t_1 = \langle x_3, x_2, x_1 \rangle$, os quais compartilham uma aresta $e = \langle x_1, x_2 \rangle$, Figura 3.8(a). Após a aplicação desse operador na aresta compartilhada e , os triângulos t_0 e t_1 passam a compartilhar a aresta $e' = \langle x_0, x_3 \rangle$, resultando nos triângulos $t_0 = \langle x_0, x_1, x_3 \rangle$ e $t_1 = \langle x_3, x_2, x_0 \rangle$, Figura 3.8(b).

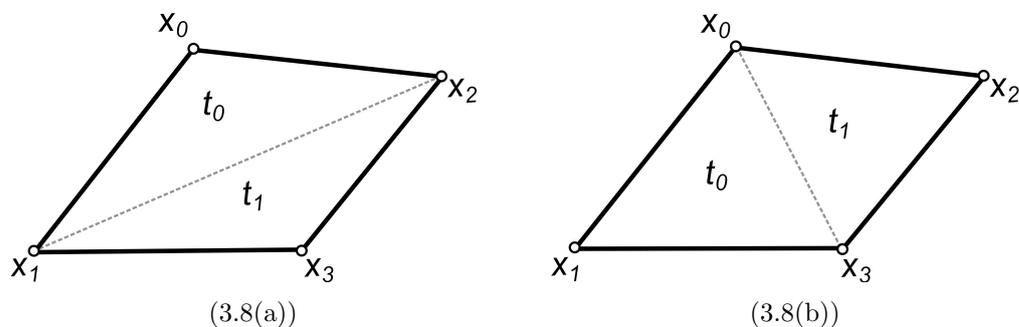


Figura 3.8: Exemplo da utilização do operador *Flipping* de aresta.

Apesar de apresentar bons resultados, esse operador deve ser utilizado mediante alguma condição de melhoria. Caso contrário, sua aplicação pode resultar na distorção dos triângulos envolvidos, Figura 3.9.

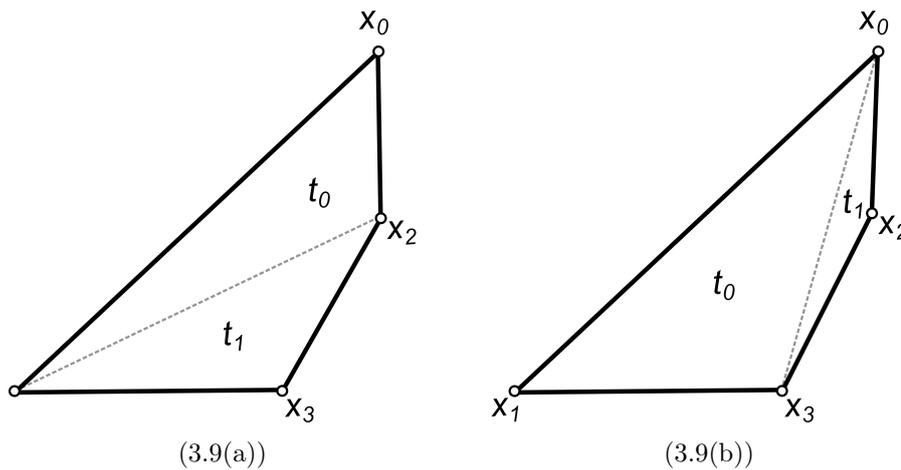


Figura 3.9: Exemplo da utilização inadequada do operador *Flipping* de aresta.

Similar ao critério apresentado no trabalho (9), o operador *Flipping* de arestas é utilizado se os ângulos internos $\theta_1, \theta_2, \theta_3, \theta_4$ do quadrilátero $Q = \langle x_0, x_1, x_2, x_3 \rangle$ satisfizerem a seguinte condição: $\theta_1 + \theta_4 \gg \theta_2 + \theta_3$, Figura 3.10.

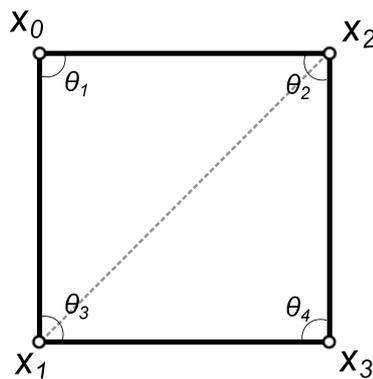


Figura 3.10: Critério de utilização do *Flipping* de aresta.

3.2 Conversão em uma malha de quadriláteros

O método proposto neste trabalho para geração de uma malha de quadriláteros a partir de uma malha composta de elementos triangulares foi inspirado no método proposto por Velho (21). Em síntese, o algoritmo inicia com a formação do conjunto de pares de triângulos adjacentes Q , os quais são chamados de *clusters*. Como não é sempre possível encontrar um pareamento perfeito, os triângulos que não fazem parte de nenhum *cluster* são chamados de triângulos isolados, os quais formam o conjunto T . Após essa classificação, os triângulos são subdivididos através de uma estratégia híbrida, em que cada elemento $T \cup Q$ é subdividido de acordo com o grupo que pertence. Essa etapa

garante que a malha em questão seja coberta por pares de triângulos, os quais são utilizados para formar os quadriláteros da malha final através da remoção da aresta compartilhada.

Na versão original do método, os *clusters* são formados através de um processo iterativo no conjunto de arestas. Para cada aresta e da malha de triângulos em ordem decrescente de tamanho, cada *cluster* é formado pelo par de triângulos adjacentes à e . A aresta compartilhada de cada *cluster* é posteriormente removida durante a geração dos quadriláteros. Portanto, para preservar as arestas que representam restrições no domínio, essas devem ser ignoradas durante a formação dos *clusters*.

O critério de formação dos *clusters* por tamanho da aresta compartilhada garante apenas que os *clusters* formados sejam convexos (21), o que não é suficiente para garantir a qualidade dos quadriláteros formados por tais *clusters*. Um exemplo dessa estratégia de formação, por ordem de tamanho da aresta compartilhada, pode ser observado na Figura 3.11, em que cada *cluster* formado é destacado nas cores verde e azul: verde para *clusters* com boa qualidade, e azul caso contrário. As arestas destacadas em vermelho representam restrições no domínio e, portanto, não foram utilizadas na formação de nenhum *cluster*. Ao analisar esse exemplo, pode-se perceber a concentração de *clusters* de baixa qualidade e triângulos isolados nas regiões próximas às fronteiras e restrições no domínio. Essa concentração ocorre porque os *clusters* formados em tais regiões não mantêm a orientação definida pelas restrições nas proximidades.

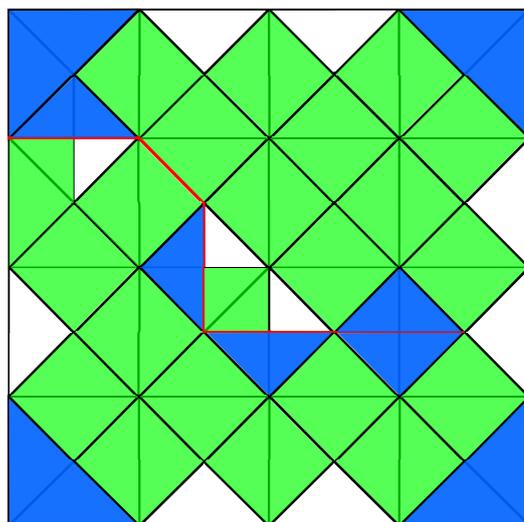


Figura 3.11: Exemplo de formação dos *clusters* de acordo com o método original (21).

Com objetivo de garantir uma qualidade mínima dos *clusters* formados e minimizar o número de triângulos isolados, o presente trabalho propõe uma

estratégia diferente para a formação dos *clusters*.

Cada triângulo da malha possui um índice que identifica o nível atual de refinamento, o qual é iniciado com 0 (malha inicial) e incrementado a cada subdivisão. Dessa forma, o nível de refinamento h_k de um triângulo k , resultado da subdivisão de um triângulo j , é dado por $h_k = h_j + 1$.

Em uma malha inicial com todos os triângulos com índice de refinamento 0 é possível obter um pareamento perfeito, considerando a malha inicial com topologia *union-and-jack*, como ilustrado na Figura 3.12(a). Nesse caso, o conjunto de *clusters* formado mantém a orientação definida pelas fronteiras do domínio. Diferente do conjunto obtido em uma malha com todos os triângulos com nível de refinamento igual a 1, Figura 3.12(b). Nesse caso, além da existência de triângulos isolados, os *clusters* formados não mantêm a orientação definida pelas fronteiras do domínio. Ao realizar mais uma iteração de subdivisão em toda a malha, resultando em triângulos com nível de refinamento 2, as características existentes na malha inicial (nível de refinamento 0) são restauradas, Figura 3.12(c). Com isso, observa-se a alternância da orientação dos *clusters* entre cada nível de refinamento, além da redução do número de triângulos isolados quando os *clusters* formados respeitam à orientação das restrições, nesse caso, restrições de fronteira.

Como a malha inicial utilizada neste trabalho não possui restrições em seu interior, idealmente os blocos básicos formados devem possuir a mesma orientação das restrições de fronteira. Para garantir um pareamento perfeito em uma malha inicial sem restrições, este trabalho utiliza uma malha inicial 4-8 com topologia *union-and-jack*.

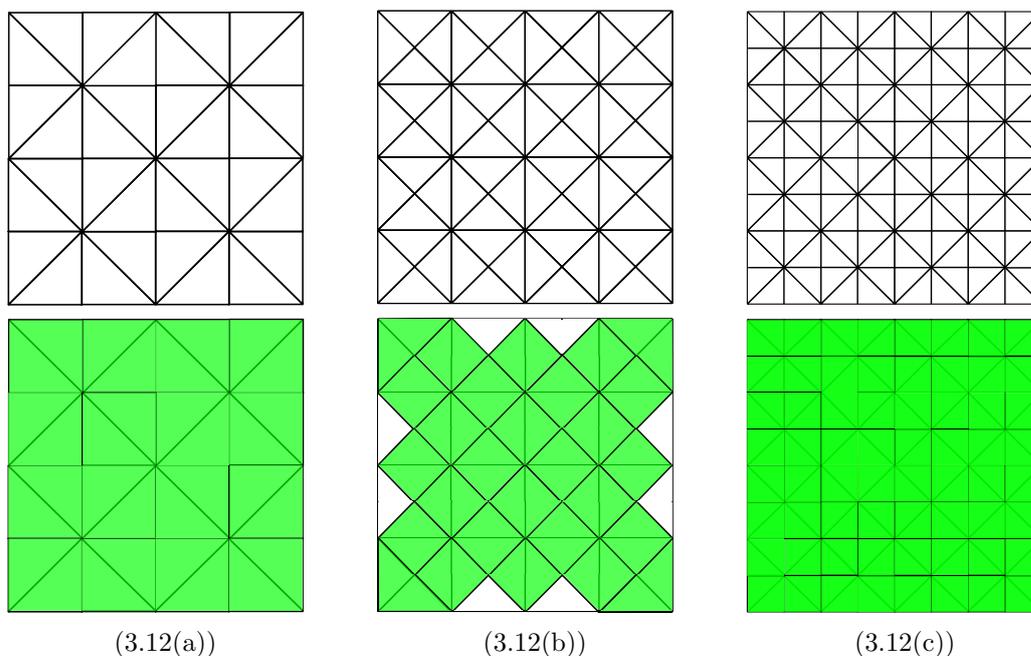


Figura 3.12: Resultado da formação dos *clusters* para diferentes níveis de refinamento. (a) Triângulos com nível 0; (b) triângulos com nível 1; (c) triângulos com nível 2.

Com o objetivo de maximizar o número de *clusters* formados nas regiões próximas às restrições, o método proposto realiza uma etapa de pré-processamento na malha. Uma visão geral dessa etapa é apresentada no Algoritmo 2. A condição existente na Linha 3 garante que os *clusters* formados nas regiões distantes das restrições mantenham a orientação definida pelas fronteiras, tentando minimizar o número de triângulos isolados. Já a condição na Linha 6, garante que a maior aresta de um triângulo não seja fixa ou de fronteira. Essa condição é criada com o objetivo de maximizar o número de *clusters* formados nas regiões próximas às restrições e fronteiras, dado que um *cluster* é formado se a aresta compartilhada é livre e interna.

Algoritmo 2: Pré-processamento da malha de triângulos.

```

1 for  $t \in T$ , onde  $T$  consiste no conjunto de triângulos da malha do
2   |  $t_2 \leftarrow$  triângulo adjacente a maior aresta de  $t$  ;
3   | if  $h_t \bmod 2 \neq 0$  and  $h_t == h_{t_2}$  then
4   |   | Aplicar operador de subdivisão na maior aresta de  $t$  ;
5   |   end
6   | if maior aresta de  $t$  for fixa ou de fronteira then
7   |   | Aplicar operador de subdivisão na maior aresta de  $t$  ;
8   |   end
9 end

```

Em seguida, a formação dos *clusters* é realizada através de um processo iterativo no conjunto de triângulos, em ordem decrescente do nível de refinamento. Essa ordem define que a formação inicie nas regiões próximas às restrições, cujos triângulos geralmente possuem maior nível de refinamento devido ao processo de adaptação da malha. Sendo assim, para cada triângulo t , em ordem decrescente do nível de refinamento, um *cluster* será formado por t e o triângulo adjacente à sua maior aresta se tal formação satisfizer as seguintes condições:

- Nenhum dos dois triângulos fizer parte de outro *cluster*;
- O ângulo máximo entre as arestas do quadrilátero formado pelo *cluster* deve ser inferior a um limite, $\max(\theta_1, \theta_2, \theta_3, \theta_4) \leq \theta_{max}$.

Após a formação dos *clusters* e identificação dos triângulos isolados, a etapa seguinte é chamada de subdivisão híbrida. Nessa etapa, cada *cluster* formado é refinado através da subdivisão binária na aresta compartilhada, como ilustrado na Figura 3.13(a). No caso dos triângulos isolados, esses são refinados através da subdivisão baricêntrica, a qual consiste na criação de três novas faces a partir da inserção de um novo vértice no baricentro do triângulo, seguido da criação de três novas arestas conectando o novo vértice a cada um dos vértices antigos, Figura 3.13(b).

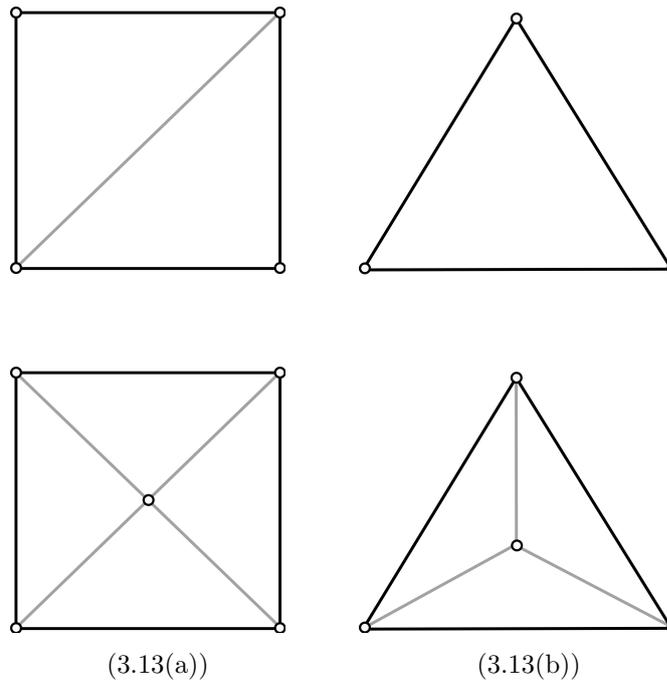


Figura 3.13: Etapa de subdivisão híbrida. (a) Subdivisão binária na aresta compartilhada do cluster; (b) subdivisão baricêntrica dos triângulos isolados.

Posteriormente, aplica-se uma iteração de subdivisão binária em cada triângulo da malha. Dessa forma, cada *cluster* será composto por quatro blocos formados por dois triângulos adjacentes, Figura 3.14(a); assim como cada triângulo isolado será composto por três blocos formados por dois triângulos adjacentes, Figura 3.14(b).

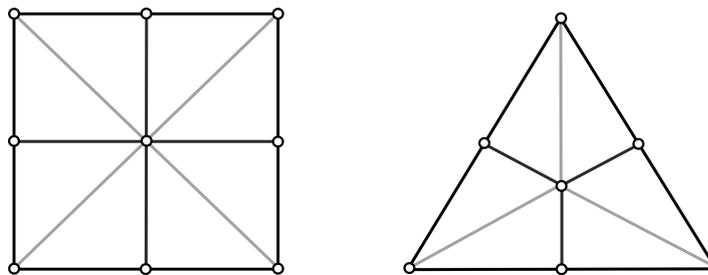


Figura 3.14: Blocos formados por pares de triângulos adjacentes.

Por fim, a aresta compartilhada de cada bloco será removida, Figura 3.15, resultando em uma malha contendo apenas elementos quadrilaterais.

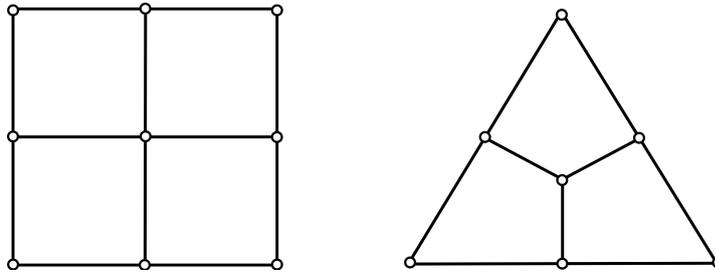


Figura 3.15: Geração dos quadriláteros através da remoção da aresta compartilhada de cada bloco.

Após a geração de uma malha composta apenas de elementos quadriláteros, o operador de suavização Laplaciano é utilizado para melhorar a qualidade da malha obtida.

4

Resultados

Este capítulo apresenta os resultados obtidos utilizando o método proposto neste trabalho. A Seção 4.1 apresenta os resultados obtidos pelo método de triangulação para um domínio complexo. Na Seção 4.2, o método de triangulação é avaliado em um caso extremo de teste. Por fim, a Seção 4.3 apresenta os resultados obtidos através do método de conversão de uma malha de triângulos em quadriláteros proposto neste trabalho.

Em todos os testes realizados, o parâmetro ϕ_{min} que define a qualidade mínima aceitável de um elemento foi definido como 0,1. O parâmetro d_{max} , o qual define a tolerância de redução de qualidade dos elementos na estrela do vértice atraído foi definido como 65%, o qual é relaxado em 65% caso nenhum dos vértices candidatos à atração satisfaça as condições existentes.

4.1

Triangulação de domínios complexos

Nesta seção, a etapa de triangulação do método proposto é avaliada através da sua aplicação em um domínio geométrico complexo. Tal domínio é composto por 252 curvas poligonais, com características baseadas em um modelo geológico real (Figura 4.1). As partes do domínio destacadas na Figura 4.2(a) podem ser observadas com maiores detalhes nas Figuras 4.2(b), (c), (d), (e), (f) e (g), possibilitando maior compreensão da complexidade do domínio em questão.

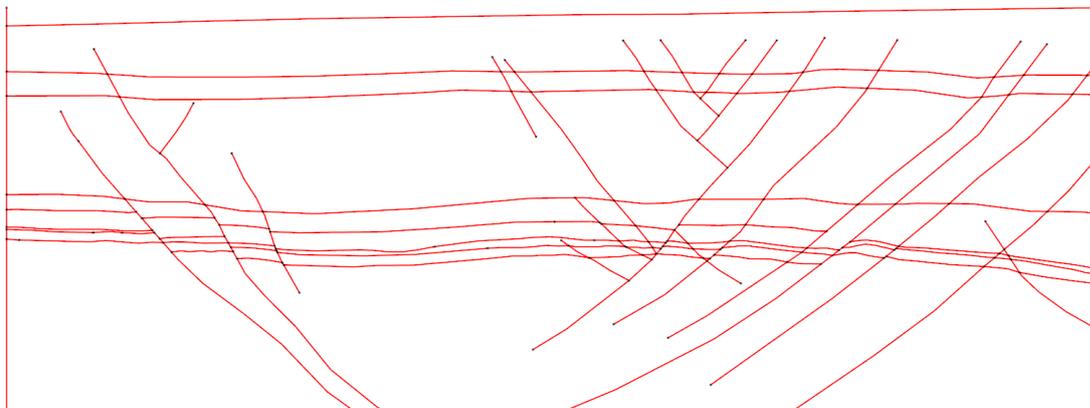
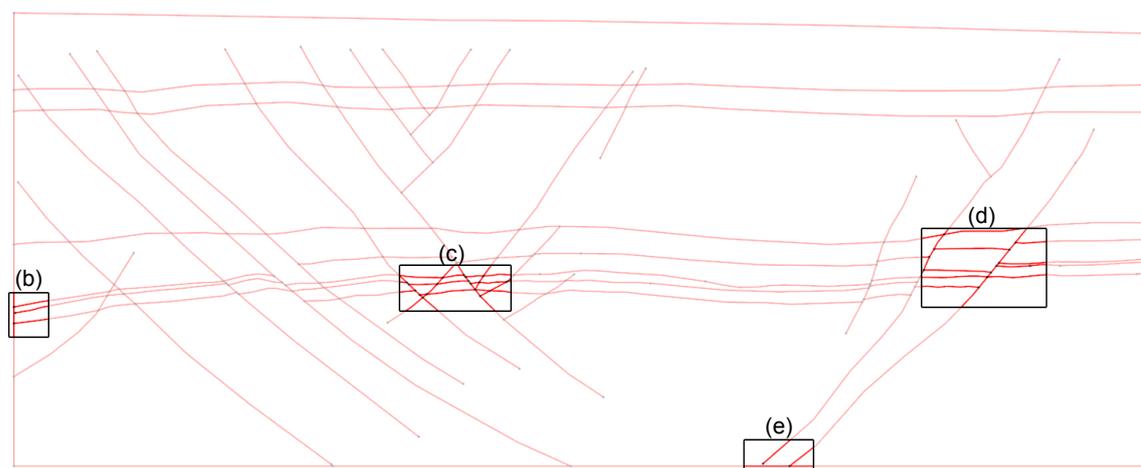
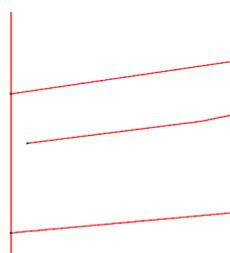


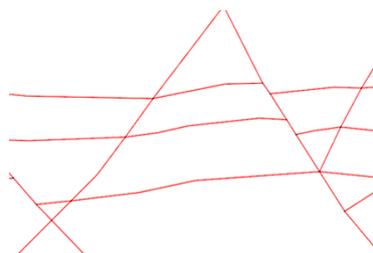
Figura 4.1: Modelo utilizado no teste.



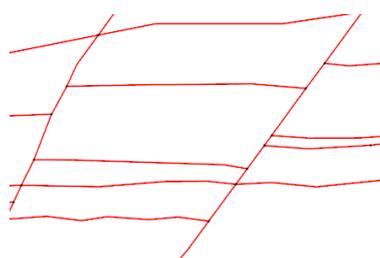
(4.2(a))



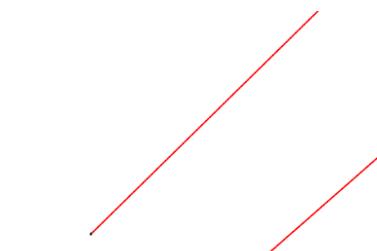
(4.2(b))



(4.2(c))



(4.2(d))



(4.2(e))

Figura 4.2: Destaque de algumas regiões do modelo utilizado nos testes.

A malha inicial utilizada como entrada no processo de adaptação é apresentada na Figura 4.3, a qual é composta por 7560 triângulos e 3913 vértices interiores. O resultado obtido para esse teste é apresentado na Figura 4.4.

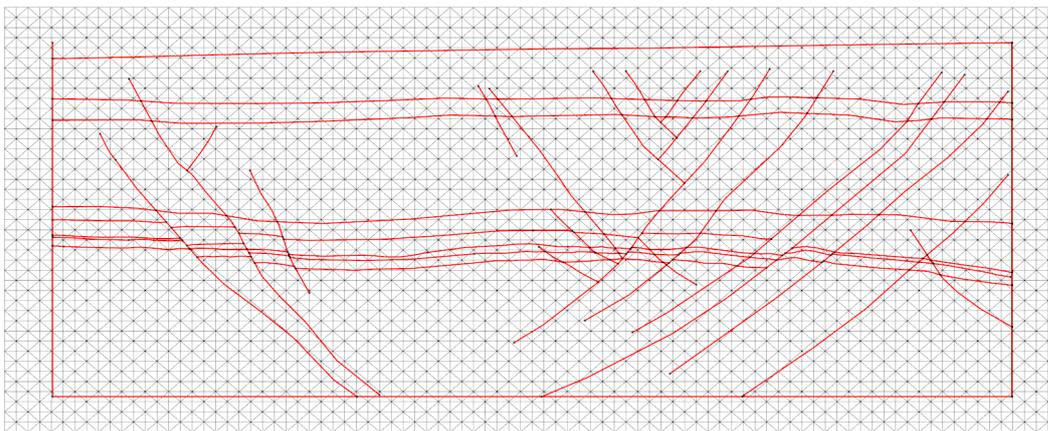


Figura 4.3: Malha inicial utilizada no teste.

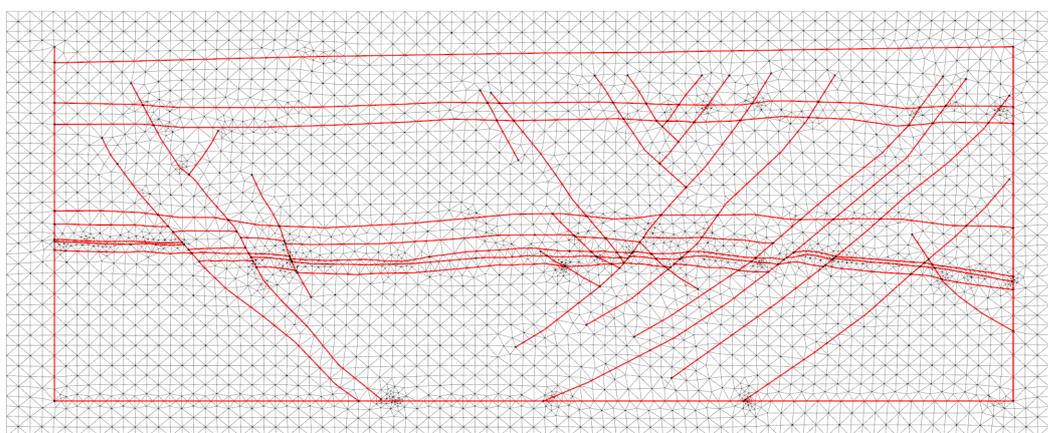
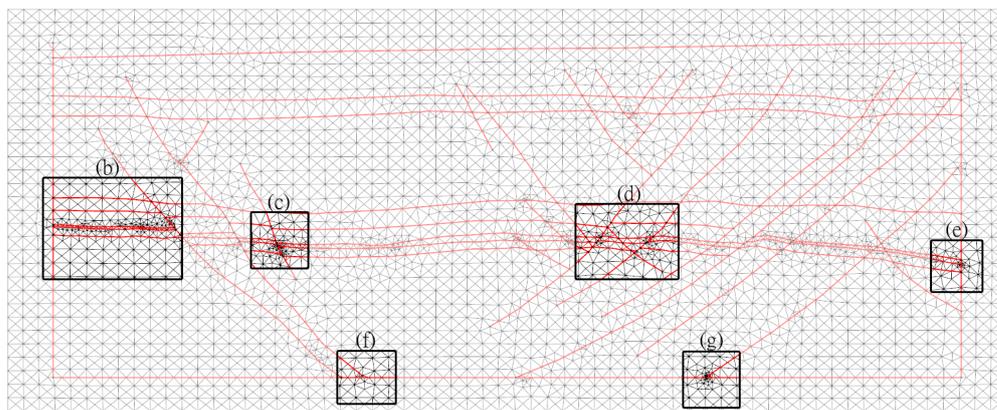
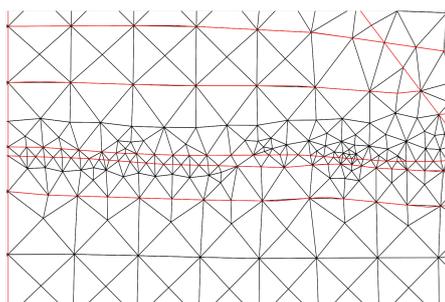


Figura 4.4: Resultado obtido.

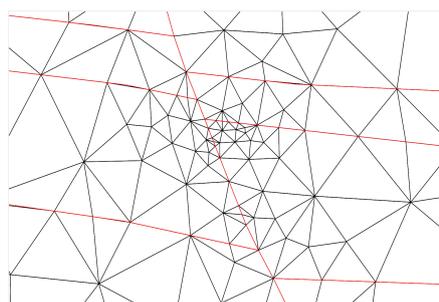
Para melhor visualizar a malha gerada, cada uma das regiões destacadas na Figura 4.5(a) é apresentada com mais detalhes nas Figuras 4.5(b), (c), (d), (e), (f) e (g).



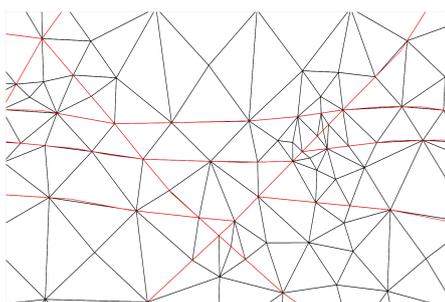
(4.5(a))



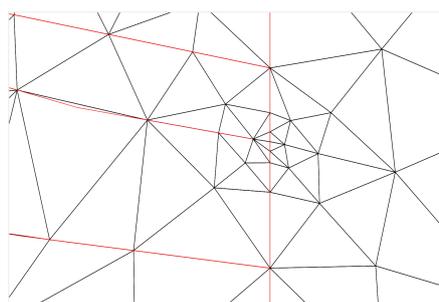
(4.5(b))



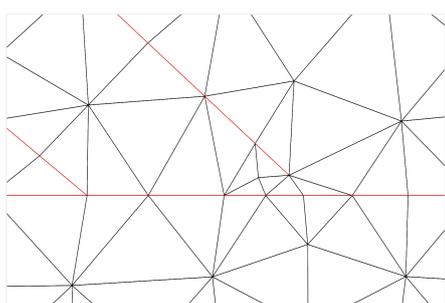
(4.5(c))



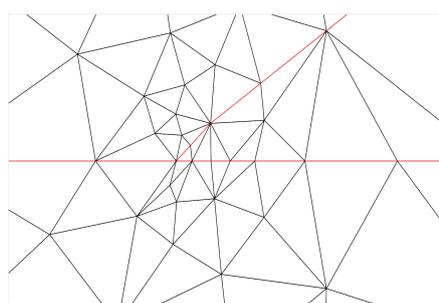
(4.5(d))



(4.5(e))



(4.5(f))



(4.5(g))

Figura 4.5: Destaque de algumas regiões do resultado obtido na triangulação.

A malha gerada é composta de 9496 triângulos e 4656 vértices interiores. A Figura 4.6 apresenta o histograma de qualidade dos triângulos, a qual é medida utilizando o critério de Lo. Tal resultado é composto de triângulos com qualidade no intervalo $[0.3, 1.0]$, sendo 97% com qualidade superior a 0.7.

Além disso, a malha é composta de vértices com valência no intervalo [4, 10], conforme ilustrado na Figura 4.7, sendo 99% com valência entre 4 e 8, seguindo a topologia da malha inicial.

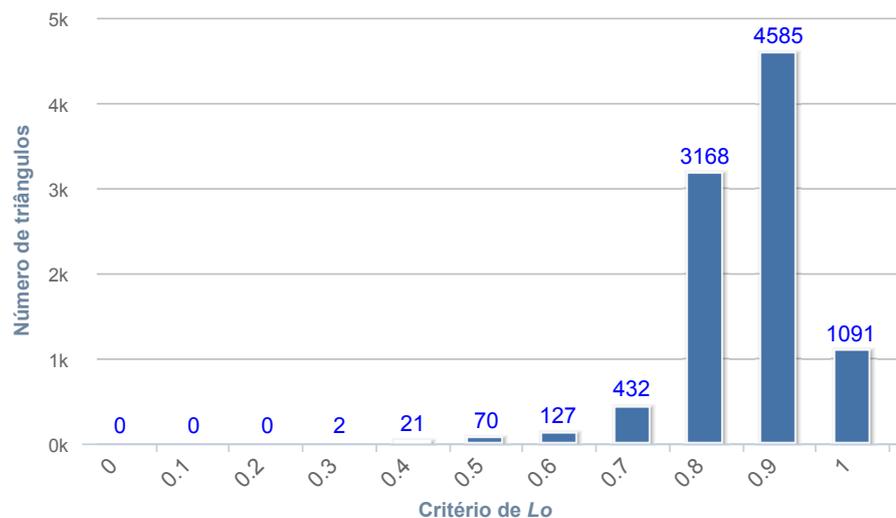


Figura 4.6: Histograma de qualidade (critério de L_0) dos triângulos.

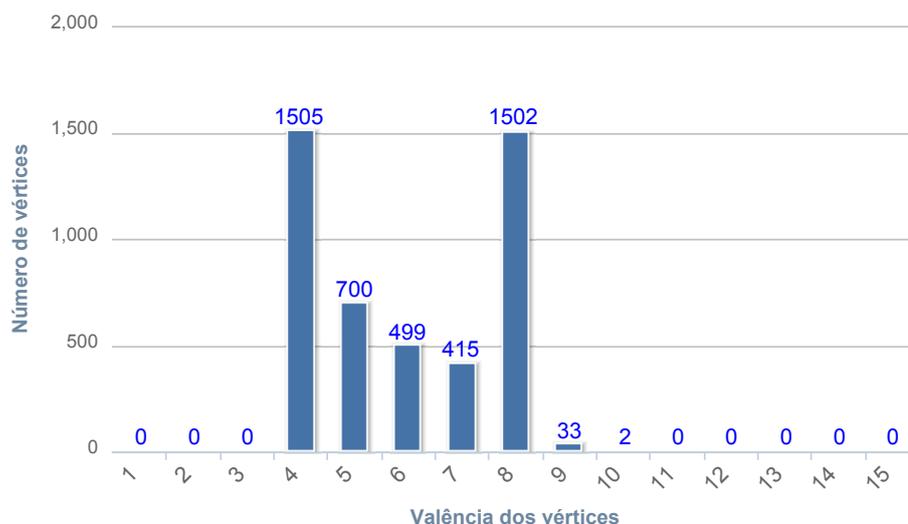


Figura 4.7: Histograma da valência dos vértices internos.

4.2 Robustez do método de triangulação

Nessa seção, o método de triangulação proposto é avaliado em um caso extremo de teste. Nesse teste, tal método é aplicado ao mesmo domínio utilizado no teste da Seção 4.1, porém a malha inicial utilizada é composta de apenas 4 triângulos e 5 vértices (Figura 4.8).

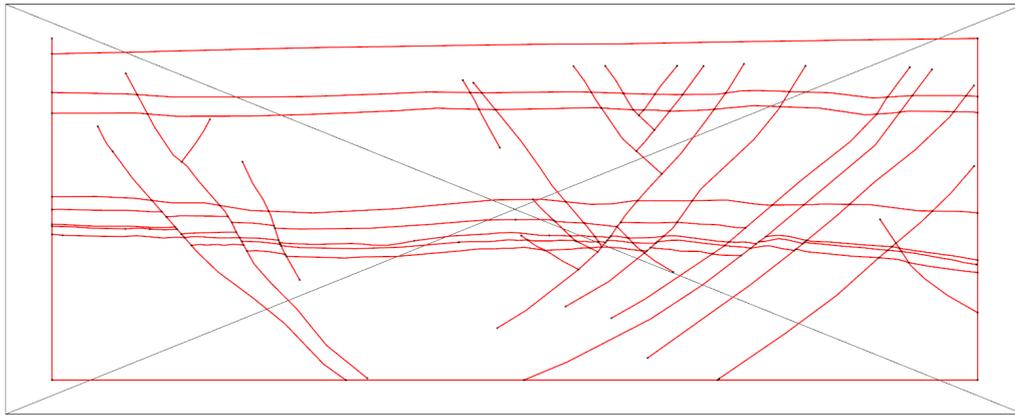


Figura 4.8: Malha Inicial utilizada no teste extremo.

A malha obtida como resultado desse teste pode ser observada na Figura 4.9, a qual é composta de 4307 triângulos e 2184 vértices internos. A qualidade dos triângulos varia no intervalo $[0.3, 1.0]$, sendo 91% dos triângulos com qualidade superior a 0.7 (Figura 4.10). Além disso, a malha é composta de vértices com valência no intervalo $[4, 11]$, sendo 94% dos vértices com valência entre 4 e 8 (Figura 4.11).

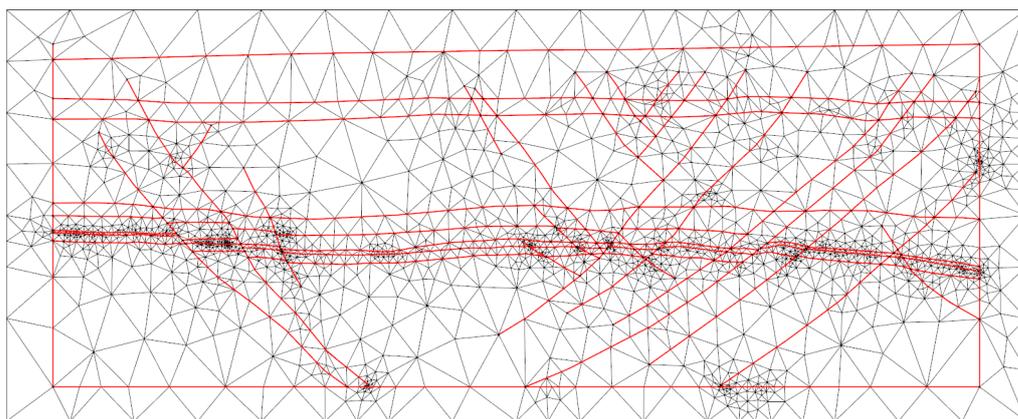


Figura 4.9: Resultado obtido no teste extremo.

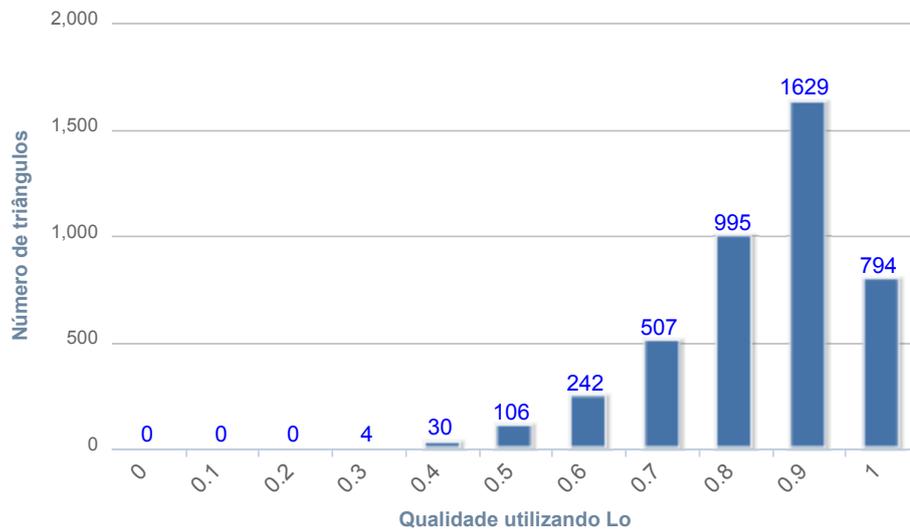


Figura 4.10: Histograma de qualidade (critério de Lo) dos triângulos que compõem a malha obtida no teste extremo.

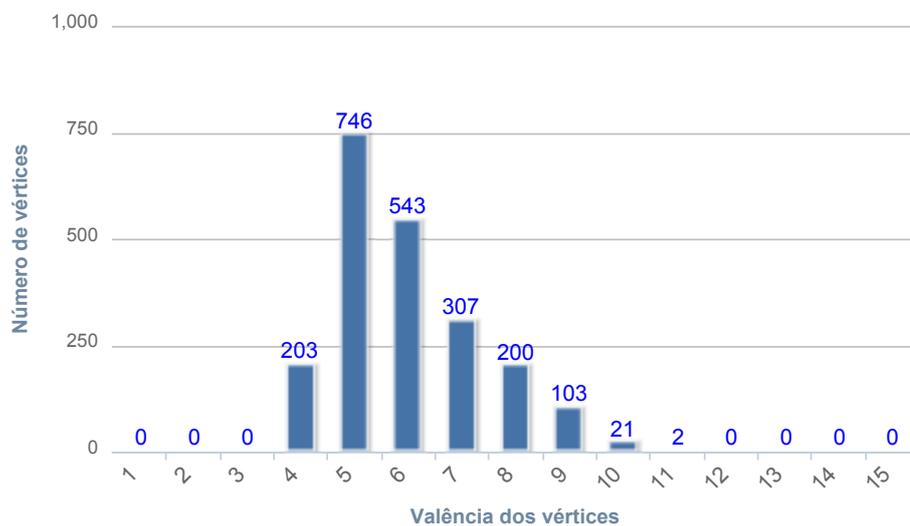


Figura 4.11: Histograma da valência dos vértices internos que compõem a malha obtida no teste extremo.

4.3

Conversão em uma malha de quadriláteros

Esta seção apresenta os resultados obtidos pelo método de conversão de uma malha de triângulos em quadriláteros proposto nesta dissertação. A qualidade da malha obtida é avaliada através do critério do ângulo mínimo de cada quadrilátero que a compõe, assim como a valência de seus vértices internos, cujo valor ideal em uma malha de quadriláteros é 4. Por fim, o método proposto é comparado com a técnica de geração de malhas utilizada no sistema Sigma2D, desenvolvido pela Tecgraf/Puc-Rio em parceria com a

PETROBRAS, a qual é baseada em métodos convencionais de mapeamento e avanço de fronteiras.

4.3.1

Aplicação em um domínio complexo

O domínio utilizado como teste nessa seção consiste no modelo apresentado na Figura 4.1. A malha de triângulos obtida como resultado do teste da Seção 4.1, a qual é apresentada na Figura 4.4, é utilizada como a triangulação de entrada para o método de conversão proposto. O resultado obtido nesse teste é apresentado na Figura 4.12.

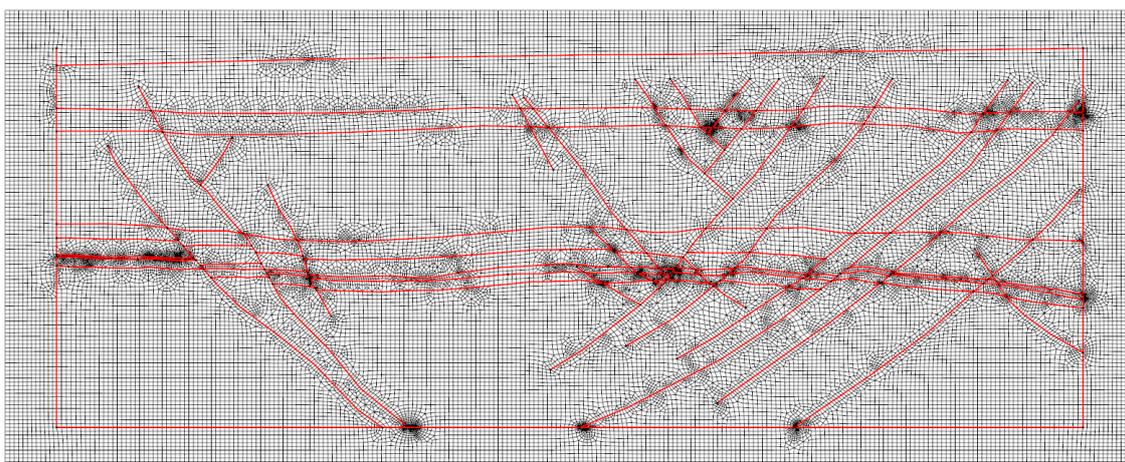
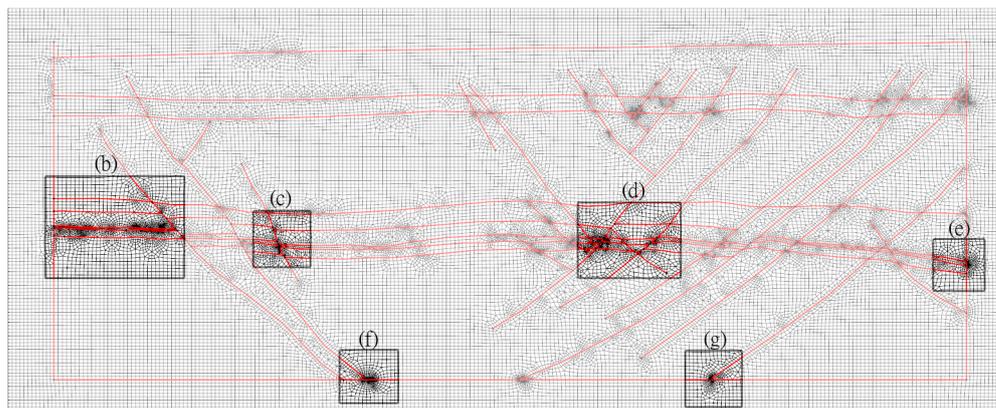
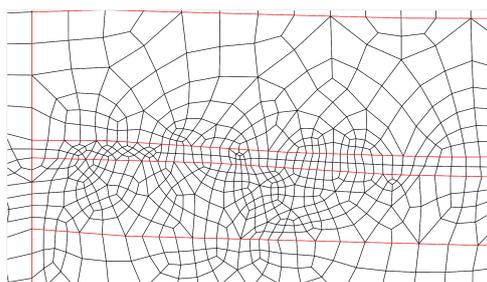


Figura 4.12: Conversão da malha de triângulos apresentada na Figura 4.4 em uma malha de quadriláteros.

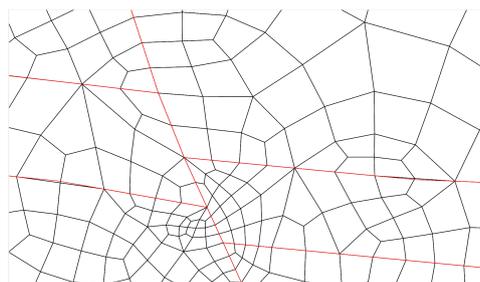
Para melhor visualizar o resultado obtido, cada uma das regiões destacadas na Figura 4.13(a) é apresentada com mais detalhes nas Figuras 4.13(b), (c), (d), (e), (f) e (g).



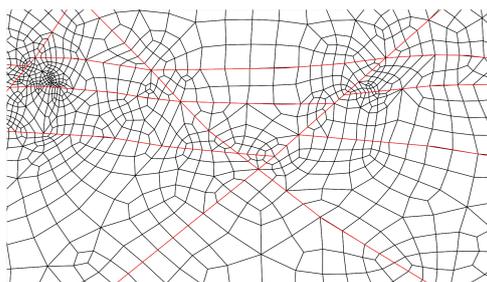
(4.13(a))



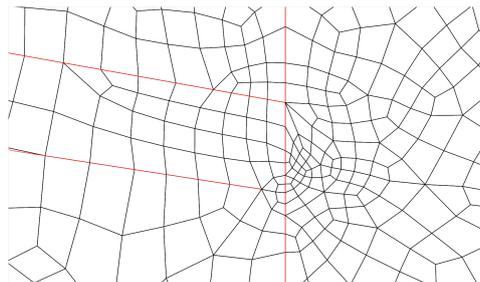
(4.13(b))



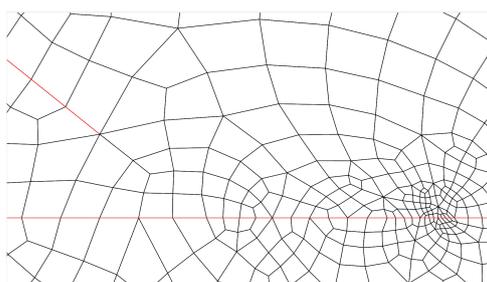
(4.13(c))



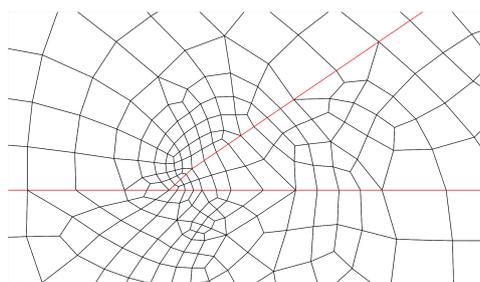
(4.13(d))



(4.13(e))



(4.13(f))



(4.13(g))

Figura 4.13: Destaque de algumas regiões do resultado obtido da conversão da malha de triângulos em quadriláteros.

A malha de quadriláteros obtida através do método de conversão é composta de 45232 elementos e 45605 vértices. O ângulo mínimo formado entre as arestas que definem cada quadrilátero da malha varia no intervalo $[18.34^\circ, 90.00^\circ]$ (Figura 4.14), sendo 68% dos quadriláteros com ângulo mínimo

acima de 70° . Os vértices da malha possuem valência no intervalo $[3, 8]$ (Figura 4.15), com 98% dos vértices internos apresentando valência entre $[3, 5]$ e 87% com valência igual a 4.

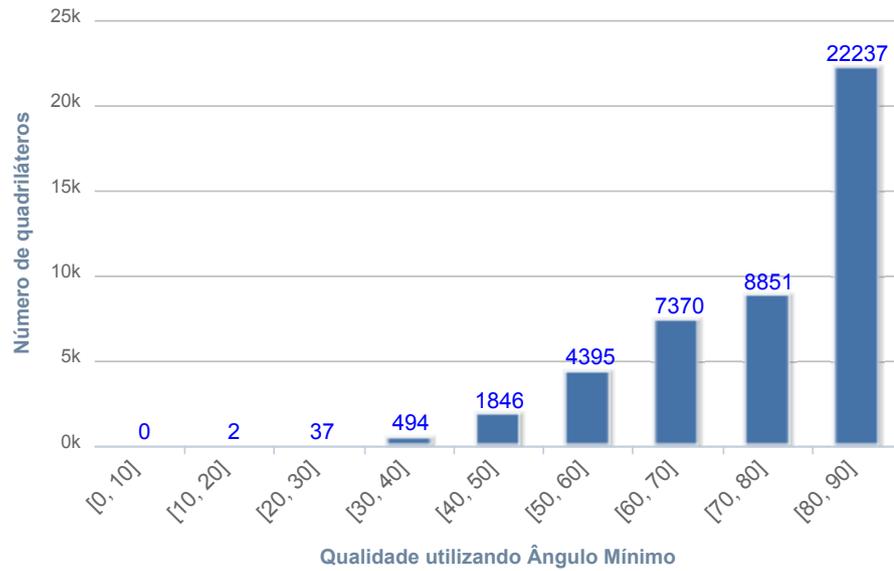


Figura 4.14: Histograma de qualidade (ângulo mínimo) dos quadriláteros.

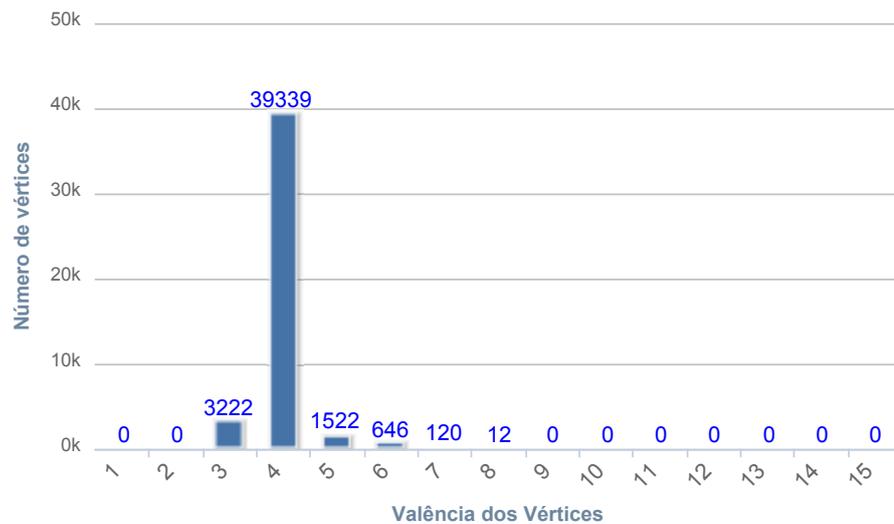


Figura 4.15: Histograma da valência dos vértices internos que compõem a malha de quadriláteros obtida.

Para avaliar a qualidade do resultado apresentado, o mesmo é comparado ao resultado obtido pelo método de geração de malhas utilizado no sistema Sigma2D, utilizando o mesmo domínio de entrada. A malha obtida como resultado desse outro método é composta de 45082 quadriláteros e 47613 vértices (Figura 4.16). Os ângulos mínimos dos quadriláteros que a compõem variam no intervalo $[13.55^\circ, 90.00^\circ]$ (Figura 4.17), sendo 66% com ângulo mínimo superior a 70° . Os vértices da malha apresentam valência no intervalo

[3, 8] (Figura 4.18), sendo 98% dos vértices com valência no intervalo [3, 5] e 81% com valência igual a 4.

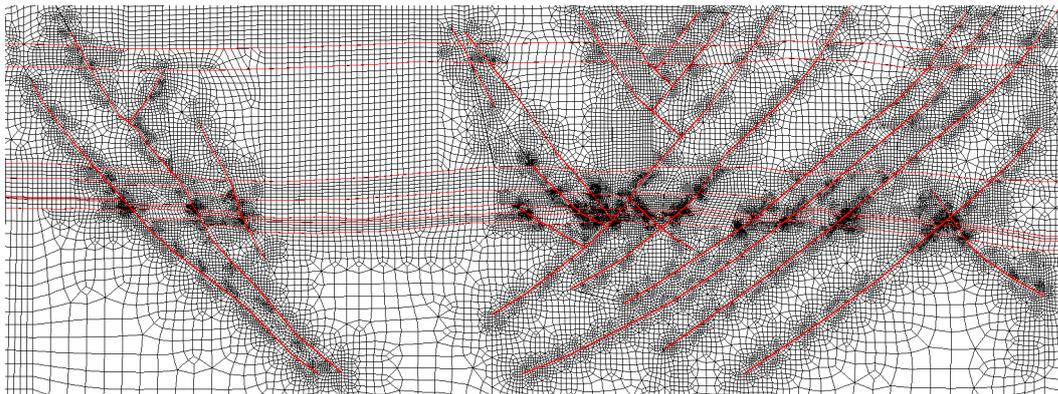


Figura 4.16: Malha de quadriláteros obtida pelo método de geração utilizado Sigma2D.

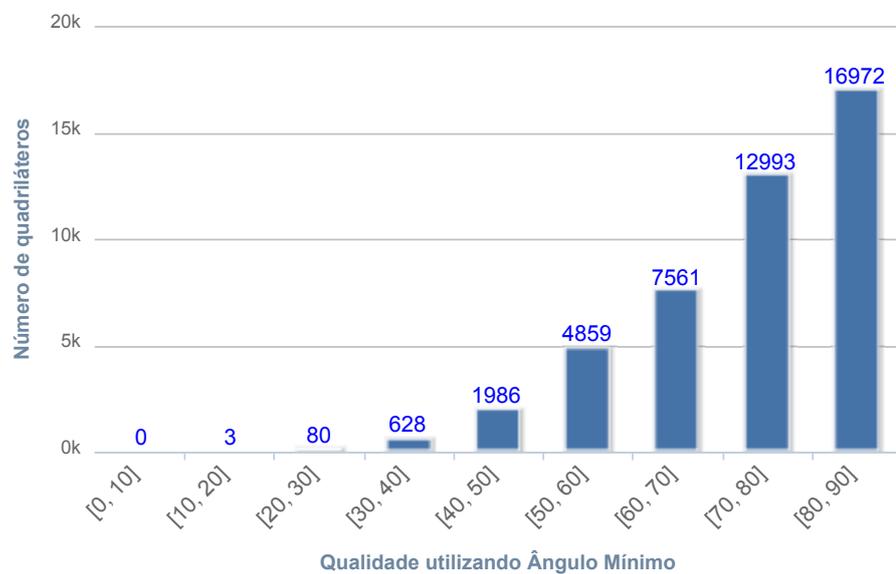


Figura 4.17: Histograma de qualidade (ângulo mínimo) da malha de quadriláteros obtida através do Sigma2D.

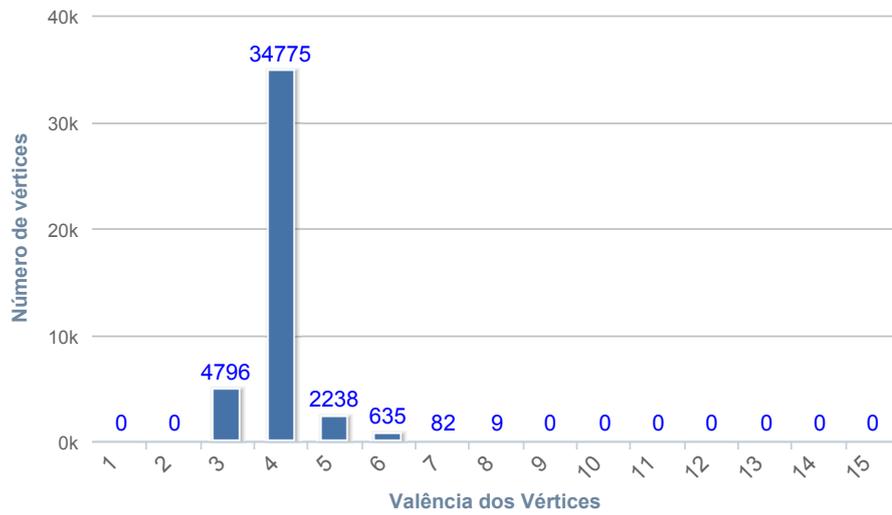


Figura 4.18: Histograma da valência dos vértices internos da malha de quadriláteros obtida através do Sigma2D.

Ao analisar os resultados obtidos por ambos os métodos, nota-se uma similaridade entre os dados apresentados para os dois critérios analisados. Além disso, o ângulo mínimo do quadrilátero com pior qualidade da malha obtida através do método proposto é de 18.35° , sendo relativamente superior a 13.55° apresentado pelo método utilizado no Sigma2D.

5 Conclusão

Esta dissertação apresentou um método automático para geração de malhas quadrilaterais. Por se tratar de um método indireto, esse método também pode ser utilizado em aplicações em que o domínio geométrico deve ser discretizado em uma malha de triângulos. Diferente da Triangulação de Delaunay, o método de triangulação apresentado não exige a definição de um conjunto de pontos distribuídos sobre o domínio. No método proposto, uma malha inicial é definida englobando todo o domínio, e as restrições que o compõem são inseridas a posteriori. A cada inserção de uma restrição, a malha existente é adaptada conforme necessário, através da utilização de operadores locais. Para converter a malha de triângulos em uma malha composta de elementos quadrilaterais, este trabalho apresenta uma variação do método de conversão proposto por Velho (21). Nesse método, os quadriláteros são gerados através da combinação de técnicas de pareamento e subdivisão dos triângulos existentes.

Os resultados apresentados mostram que o método proposto nesta dissertação obtém resultados de qualidade para malhas compostas de elementos triangulares ou quadrilaterais, mesmo quando aplicado a domínios complexos. Além disso, especula-se que esse método apresenta complexidade linear em relação ao número de vértices da malha gerada e em relação à discretização das restrições do domínio, uma vez que esse método é composto de operadores topológicos locais, cujo custo não varia em relação ao tamanho do problema. Contudo, uma análise de complexidade criteriosa é deixada como trabalho futuro.

Além disso, o algoritmo de triangulação proposto mostra-se adequado na adaptação de uma malha de triângulos existente a novas restrições. Sendo assim, a aplicação de tal método na inserção de fraturas em uma malha é automática, uma vez que as arestas alinhadas a uma determinada restrição consistem nas arestas a serem fraturadas. Além disso, a estrutura TopS utilizada já dá suporte a esse tipo de operação.

Por fim, pretende-se investigar a extensão do método proposto para domínios em três dimensões. Acredita-se que o principal desafio apresentado

por essa extensão seja a conversão de uma malha de tetraedros em hexaedros.

6

Referências Bibliográficas

- [1] BAEHMANN, P. L.; WITTCHEN, S. L.; SHEPHARD, M. S.; GRICE, K. R. ; YERRY, M. A. Robust, geometrically based, automatic two-dimensional mesh generation, **International Journal for Numerical Methods in Engineering**, v.24, n.6, p. 1043–1078, 1987.
- [2] BERN, M.; PLASSMANN, P. **Mesh generation**. Em: HANDBOOK OF COMPUTATIONAL GEOMETRY. ELSEVIER SCIENCE, p. 291–332, 2000.
- [3] BLACKER, T. D.; STEPHENSON, M. B. Paving: A new approach to automated quadrilateral mesh generation, **International Journal for Numerical Methods in Engineering**, v.32, n.4, p. 811–847, 1991.
- [4] CATMULL, E.; CLARK, J. **Seminal graphics**. chapter Recursively generated B-spline surfaces on arbitrary topological meshes, p. 183–188. ACM, New York, NY, USA, 1998.
- [5] CELES, W.; PAULINO, G. H. ; ESPINHA, R. A compact adjacency-based topological data structure for finite element mesh representation, **International Journal for Numerical Methods in Engineering**, v.64, n.11, p. 1529–1556, 2005.
- [6] CHEW, L. P. **Constrained delaunay triangulations**. Em: PROCEEDINGS OF THE THIRD ANNUAL SYMPOSIUM ON COMPUTATIONAL GEOMETRY, p. 215–222, New York, NY, USA, 1987. ACM.
- [7] DESBRUN, M.; MEYER, M.; SCHRODER, P. ; BARR, A. H. **Implicit fairing of irregular meshes using diffusion and curvature flow**. Em: PROCEEDINGS OF THE 26TH ANNUAL CONFERENCE ON COMPUTER GRAPHICS AND INTERACTIVE TECHNIQUES, SIGGRAPH '99, p. 317–324, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [8] ERTEN, H.; UNGOR, A. ; ZHAO, C. **Mesh smoothing algorithms for complex geometric domains**. Em: Clark, B. W., editor, PROCEEDINGS OF THE 18TH INTERNATIONAL MESHING ROUNDTABLE, p. 175–193. Springer Berlin Heidelberg, 2009.

- [9] HUANG, C.-Y.; LAI, J.-Y. ; LIN, K.-Y. Automatic construction of quadrilateral network of curves from triangular meshes, **Advances in Engineering Software**, v.41, n.2, p. 388 – 400, 2010.
- [10] LEE, K.-Y.; KIM, I.-I.; CHO, D.-Y. ; WAN KIM, T. An algorithm for automatic 2d quadrilateral mesh generation with line constraints, **Computer-Aided Design**, v.35, n.12, p. 1055 – 1068, 2003.
- [11] LIN, W.; TANG, Y.; ZHAO, C.; LIU, X.; ZHU, G. ; JIANG, F. An algorithm for automatic 2d finite element mesh generation with line constraints, **Computer-Aided Design**, v.43, n.12, p. 1803 – 1813, 2011.
- [12] LIU, Y.; XING, H. L. ; GUAN, Z. An indirect approach for automatic generation of quadrilateral meshes with arbitrary line constraints, **International Journal for Numerical Methods in Engineering**, v.87, n.9, p. 906–922, 2011.
- [13] LO, S. Generating quadrilateral elements on plane and over curved surfaces, **Computers and Structures**, v.31, n.3, p. 421 – 426, 1989.
- [14] MOHAMED S. EBEIDA, KAAAN KARAMETE, E. M.; DEY, S. **Q-tran: A new approach to transform triangular meshes into quadrilateral meshes locally**. Em: PROCEEDINGS, 19TH INTERNATIONAL MESHING ROUNDTABLE, p. 23–34. Springer-Verlag, 2010.
- [15] NOWOTTNY, D. **Quadrilateral mesh generation via geometrically optimized domain decomposition**. Em: PROCEEDINGS, 6TH INTERNATIONAL MESHING ROUNDTABLE, SANDIA NATIONAL LABORATORIES, p. 309–320, 1997.
- [16] OWEN, S. J. **A survey of unstructured mesh generation technology**. Em: IMR, p. 239–267, 1998.
- [17] OWEN, S. J.; STATEN, M. L.; CANANN, S. A. ; SAIGAL, S. Q-morph: an indirect approach to advancing front quad meshing, **International Journal for Numerical Methods in Engineering**, v.44, n.9, p. 1317–1340, 1999.
- [18] PARK, C.; NOH, J.-S.; JANG, I.-S. ; KANG, J. M. A new automated scheme of quadrilateral mesh generation for randomly distributed line constraints, **Comput. Aided Des.**, v.39, n.4, p. 258–267, 2007.
- [19] SUAREZ, J.; PLAZA, A. Block-balanced meshes in iterative uniform refinement, **Computer Aided Geometric Design**, v.23, n.9, p. 684 – 697, 2006.

- [20] TAM, T.; ARMSTRONG, C. 2d finite element mesh generation by medial axis subdivision, **Advances in Engineering Software and Workstations**, v.13, n.5-6, p. 313 – 324, 1991.
- [21] VELHO, L. **Quadrilateral meshing using 4-8 clustering**. Em: PROCEEDINGS OF CILANCE 2000 - SYMPOSIUM ON MESH GENERATION AND SELF-ADAPTIVITY, p. 61–64, 2000.
- [22] VELHO, L. Using semi-regular 4-8 meshes for subdivision surfaces, **J. Graph. Tools**, v.5, n.3, p. 35–47, 2000.
- [23] VELHO, L.; GOMES, J. Variable resolution 4-k meshes: Concepts and applications, **Computer Graphics Forum**, v.19, n.4, p. 195–212, 2000.
- [24] VELHO, L.; ZORIN, D. 4-8 subdivision, **Computer Aided Geometric Design**, v.18, n.5, p. 397 – 427, 2001.
- [25] WHITE, D.; KINNEY, P. **Redesign of the paving algorithm: Robustness enhancements through element by element meshing**. Em: PROC. 6 TH INT. MESHING ROUNDTABLE, p. 323–335, 1997.